

Introdução

Sistemas Microcontrolados

Atendimento ao Aluno

- ▶ Paluno
 - ▶ Quarta-feira 14:40 ~ 16:20
 - ▶ Quinta-feira 9:10 ~ 10:50

Informações da disciplina

Código Ofertado	Disciplina/Unidade Curricular	Modo de Avaliação	Modalidade da disciplina	Oferta
STCO7A	Sistemas Microcontrolados	Nota/Conceito E Frequência	Presencial	Semestral

Carga Horária					
AT	AP	APS	ANP	APCC	Total
2	2	0	0	0	60

• AT: Atividades Teóricas (aulas semanais).
• AP: Atividades Práticas (aulas semanais).
• ANP: Atividades não presenciais (horas no período).
• APS: Atividades Práticas Supervisionadas (aulas no período).
• APCC: Atividades Práticas como Componente Curricular (aulas no período, esta carga horária está incluída em AP e AT).
• Total: carga horária total da disciplina em horas.

Conteúdo Programático

Ordem	Ementa	Conteúdo
1	Arquitetura: tipos de arquiteturas.	Arquiteturas Harvard e Von Neumann, estrutura interna de uma CPU e ciclos de máquina.
2	Registradores.	Estudo dos diversos registradores internos, suas funções e configurações.
3	Projeto do subsistema de memória. Modos de endereçamento.	Tipos de memórias: flash, RAM, ROM. Modos de acesso, ciclos de escrita e leitura. Endereçamento por registrador, direto, indireto, relativo, absoluto, longo e indexado.

Ordem	Ementa	Conteúdo
4	Sistema de interrupções e exceções. Temporizadores.	Estudo dos modos de interrupção, configurações e aplicações. Temporizadores, funções e configurações.
5	Comunicação serial. Barramentos e interfaces integradas	Interface de comunicação serial.
6	Periféricos e interfaces integradas.	Dispositivos de entrada e saída (I/O). Conversores A/D e D/A: características de dispositivos comerciais e interfaceamento com microprocessador. Sensores e outros dispositivos digitais.
7	Desenvolvimento de projetos utilizando microcontroladores.	Desenvolvimento de projeto completo de hardware e software utilizando microcontrolador, memória e dispositivos de I/O.

Bibliografia Básica

PEREIRA, Fábio. **Tecnologia ARM:** microcontroladores de 32 bits. 1. ed. São Paulo, SP: Érica, 2007. 448 p. ISBN 9788536501703.

SOUZA, Daniel Rodrigues de; SOUZA, David José de; LAVINIA, Nicolás César. **Desbravando o microcontrolador PIC18:** recursos avançados. 1. ed. São Paulo, SP: Érica, 2010. 336 p. ISBN 9788536502632.

VALVANO, Jonathan W. **Embedded systems:** introduction to ARM Cortex-M microcontrollers. 5th ed. [Austin, US]: University of Texas, c2017. xii, 495 p. ISBN 9781477508992 (v.1).

Bibliografia Complementar

ALMEIDA, Rodrigo Maximiano Antunes de; MORAES, Carlos Henrique Valério; SERAPHIM, Thatyana de Faria Piola. **Programação de sistemas embarcados:** desenvolvendo software para microcontroladores em linguagem C. 1. ed. Rio de Janeiro, RJ: Elsevier, 2016. xix, 467 p. ISBN 9788535285185.

BERTELS, Koen. **Hardware/Software Co-design for Heterogeneous Multi-core Platforms.** Springer, 2012. ISBN 978-9400797192

BERTOGNA, Eduardo G. **Microcontroladores AVR Teoria e Prática.** Clube de Autores. 2014.

SOUZA, Daniel R.; SOUZA, David J. **Desbravando o Microcontrolador PIC 18.** Ensino Didático. São Paulo: Érica, 2012.

VALVANO, J. W. **Embedded Systems:** Real-Time Operating Systems for Arm Cortex-M Microcontrollers. 4th ed. Austin (Texas, Estados Unidos): University of Texas, 2017. ISBN 9781466468863.

O que é um Microcontrolador

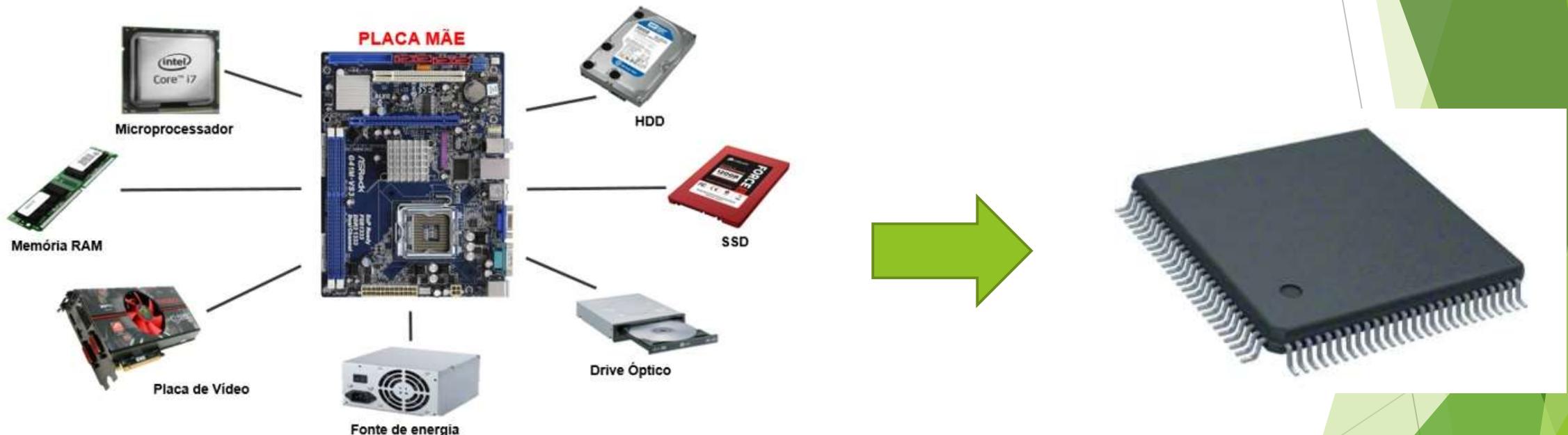
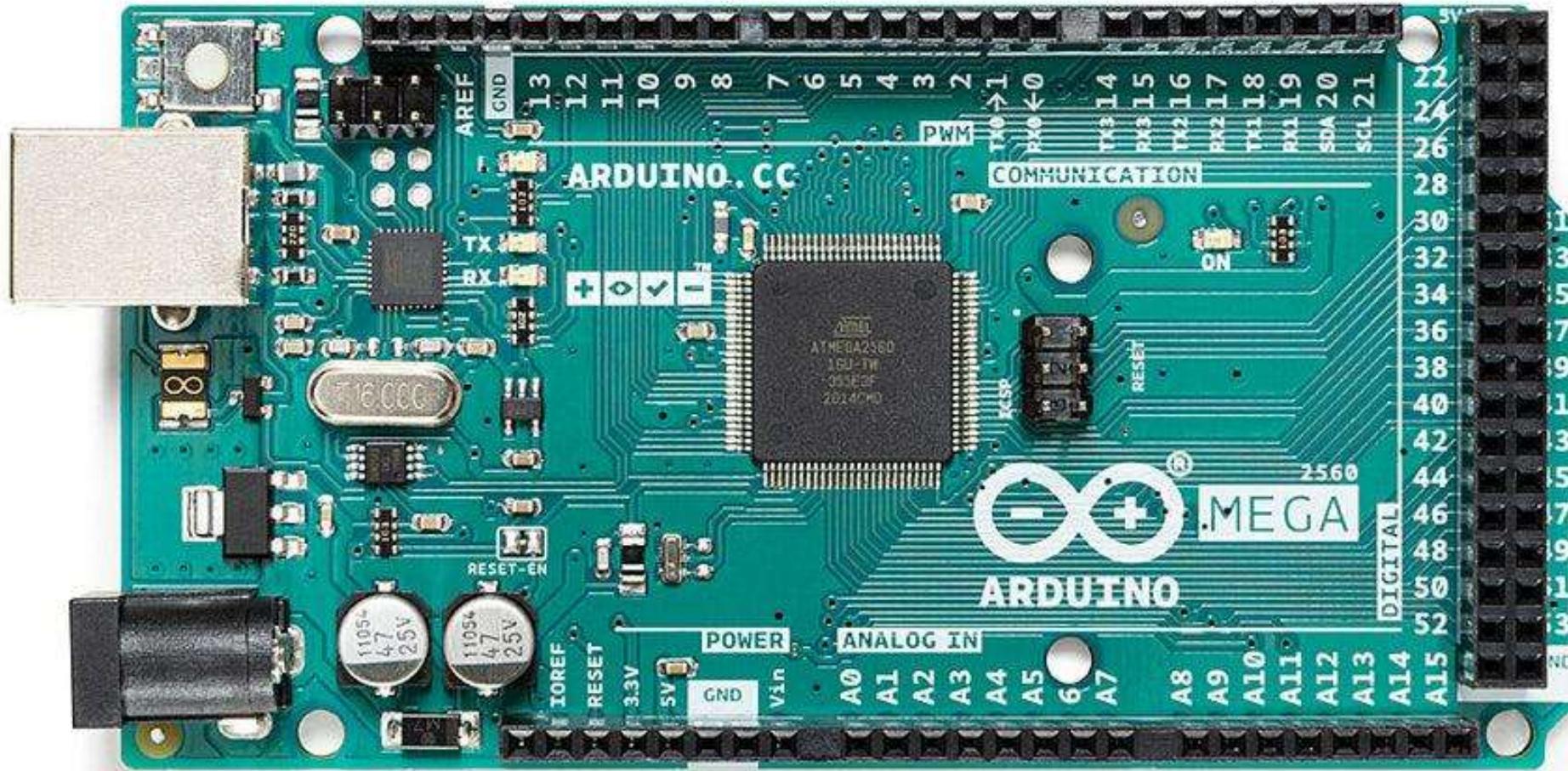


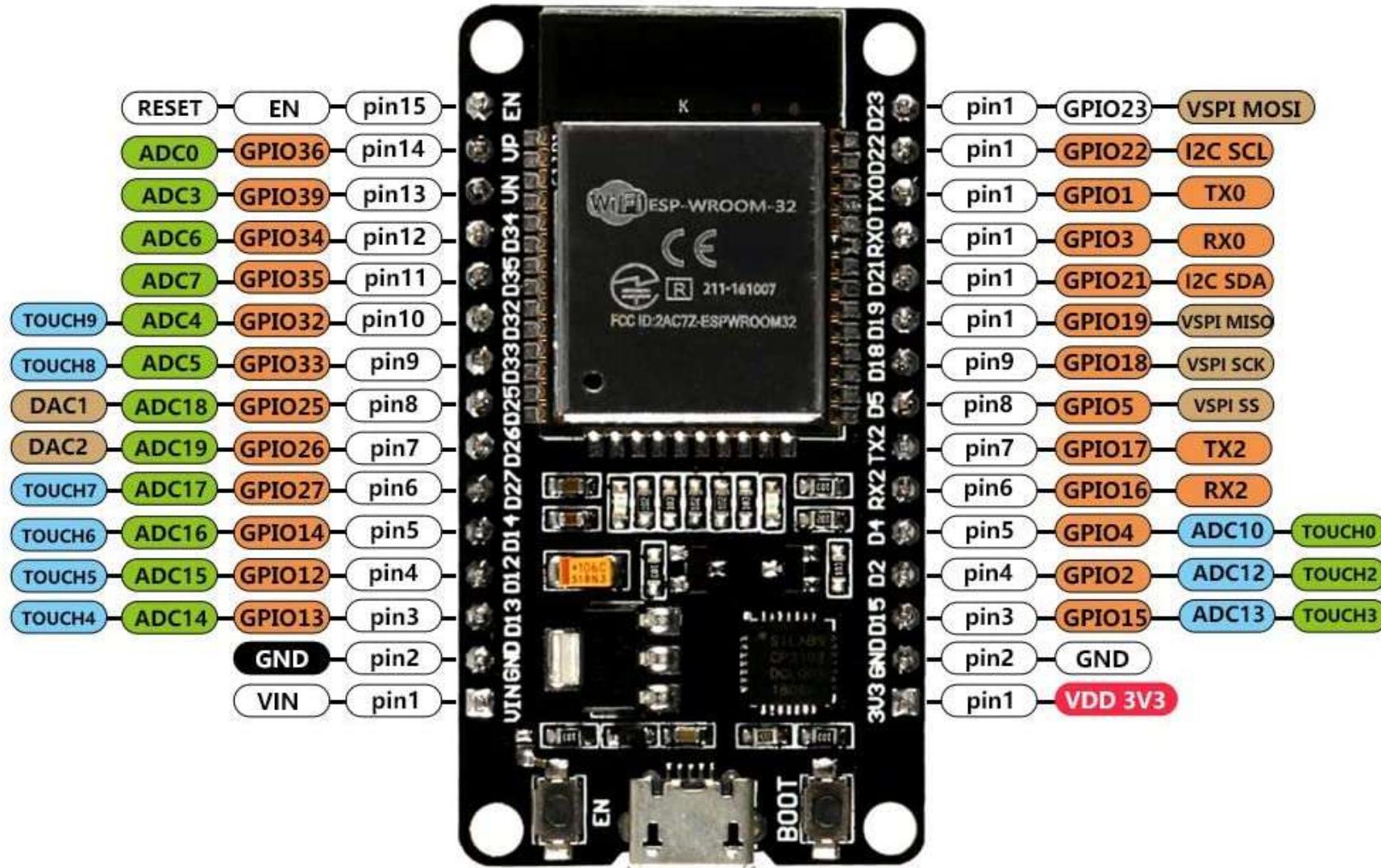
Figura: Hardwares internos do computador

Arduino Mega



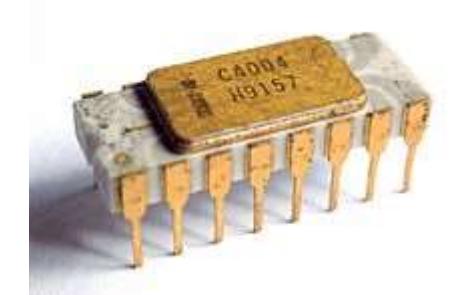
Microcontrolador Atmel2560

ESP32



História - Como chegamos até aqui

- ▶ O primeiro microprocessador (comercial) foi o [Intel 4004](#), lançado um único chip MOS LSI em 1971 (metal-oxide-semiconductor) (large-scale integration).
- ▶ Gary Boone e Michael Cochran criam o primeiro microcontrolador em 1971, o [TMS 1000](#), que tornou-se comercial em 1974. (Texas Instruments)
- ▶ Em resposta ao TMS 1000, a Intel desenvolveu um microcontrolador otimizado para aplicações de controle,o [Intel 8048](#), que tornou-se comercial em 1977.



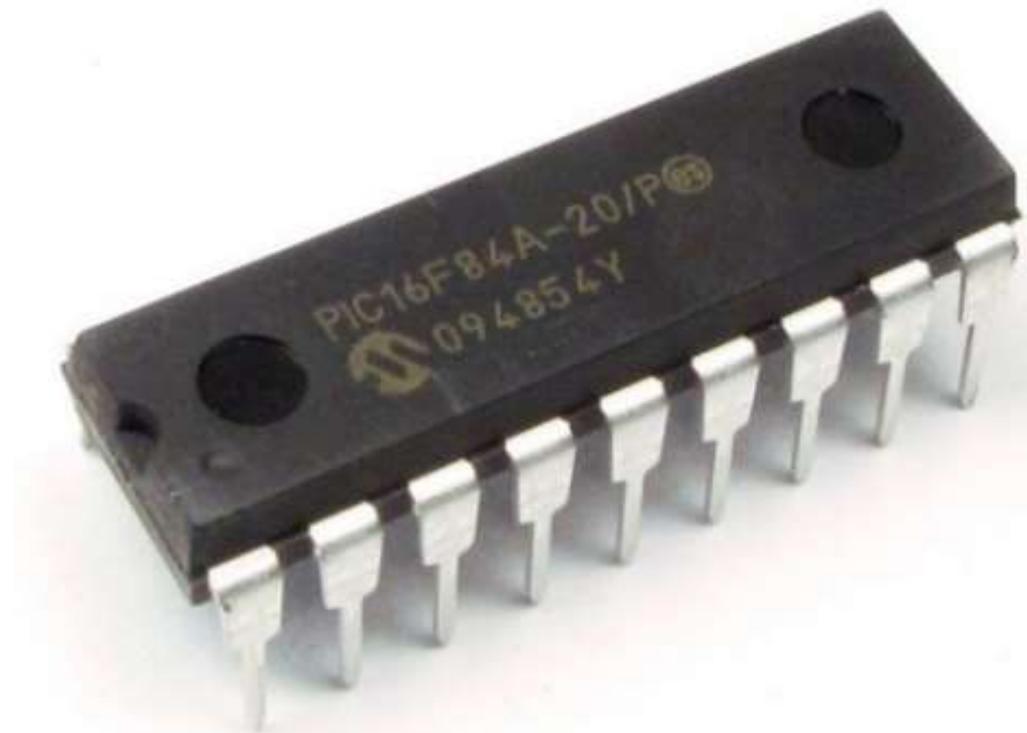
História - Chip com Janela

- ▶ ROM programável (PROM)
- ▶ One had EPROM program memory, with a transparent quartz window in the lid of the package to allow it to be erased by exposure to ultraviolet light.



História - Quase lá

- ▶ In 1993, the introduction of EEPROM memory allowed microcontrollers (beginning with the Microchip PIC16C84). (EEPROM - electrically erasable programmable read-only memory)



8051

Intel MCS-51 second sources



AMD D87C51



MHS S-80C31



OKI M80C31



Philips PCB80C31



Signetics SCN8031



Temic TS80C32

8051

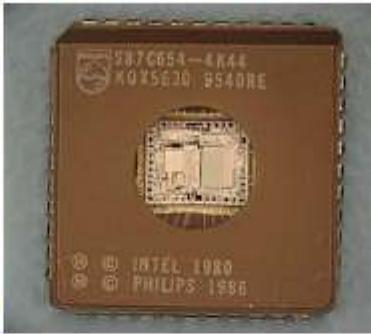
Intel MCS-51 derived microcontrollers



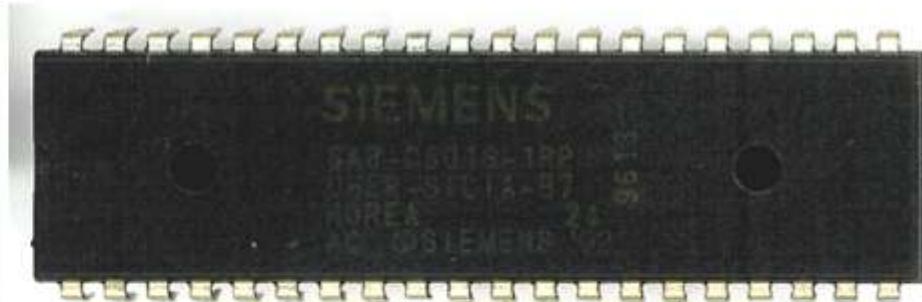
Atmel AT89C2051



Infineon SAB-C515



Philips S87C654



Siemens SAB-C501

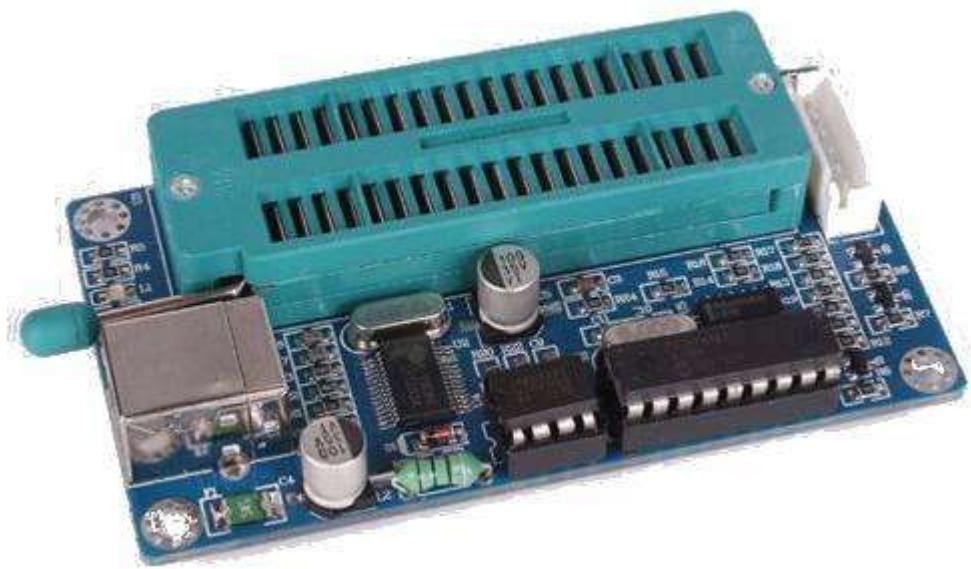


STC Micro STC89C52

Fabricantes

- | | | |
|-------------------------------|---------------------------|-----------------------|
| 1 Altera | 16 Microchip Technology | 31 STMicroelectronics |
| 2 Analog Devices | 17 National Semiconductor | 32 Texas Instruments |
| 3 Atmel | 18 NEC | 33 Toshiba |
| 4 Cypress Semiconductor | 19 NXP Semiconductors | 34 Ubicom |
| 5 ELAN Microelectronics Corp. | 20 Nuvoton Technology | 35 WCH |
| 6 EPSON Semiconductor | 21 Panasonic | 36 Xemics |
| 7 Espressif Systems | 22 Parallax | 37 Xilinx |
| 8 Freescale Semiconductor | 23 Rabbit Semiconductor | 38 XMOS |
| 9 Fujitsu | 24 Renesas Electronics | 39 ZiLOG |
| 10 Holtek | 25 Redpine Signals | 40 Sortable table |
| 11 Hyperstone | 26 Rockwell | 41 References |
| 12 Infineon | 27 Silicon Laboratories | |
| 13 Intel | 28 Silicon Motion | |
| 14 Lattice Semiconductor | 29 Sony | |
| 15 Maxim Integrated | 30 Spansion | |

Gravadores

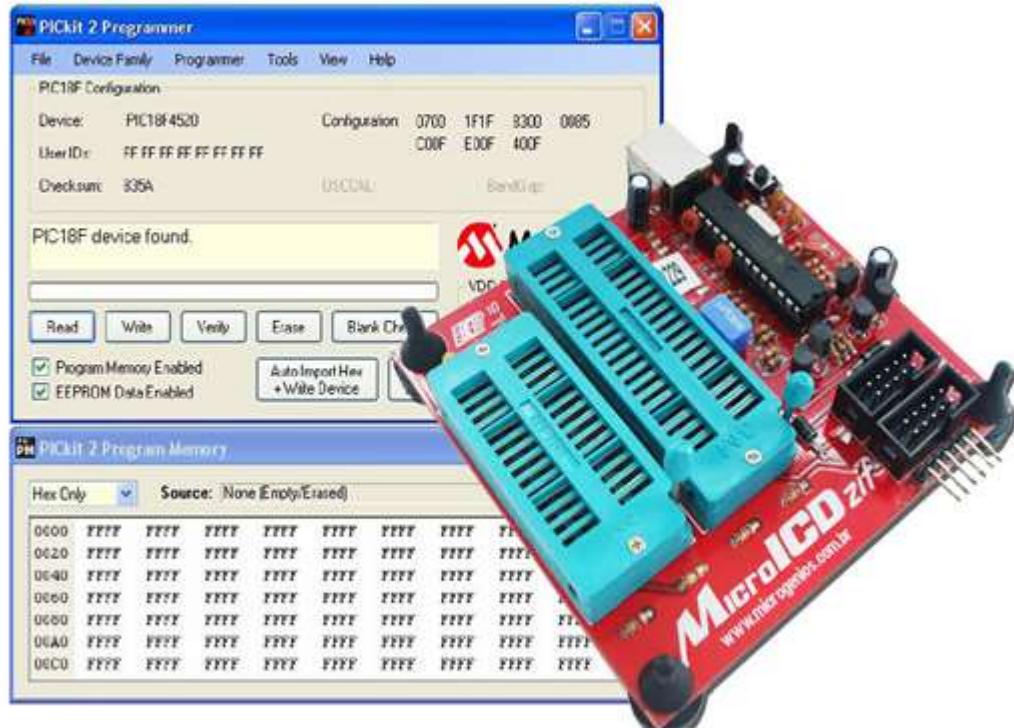


Gravador de PIC



Gravador de
ATMEL

Gravadores



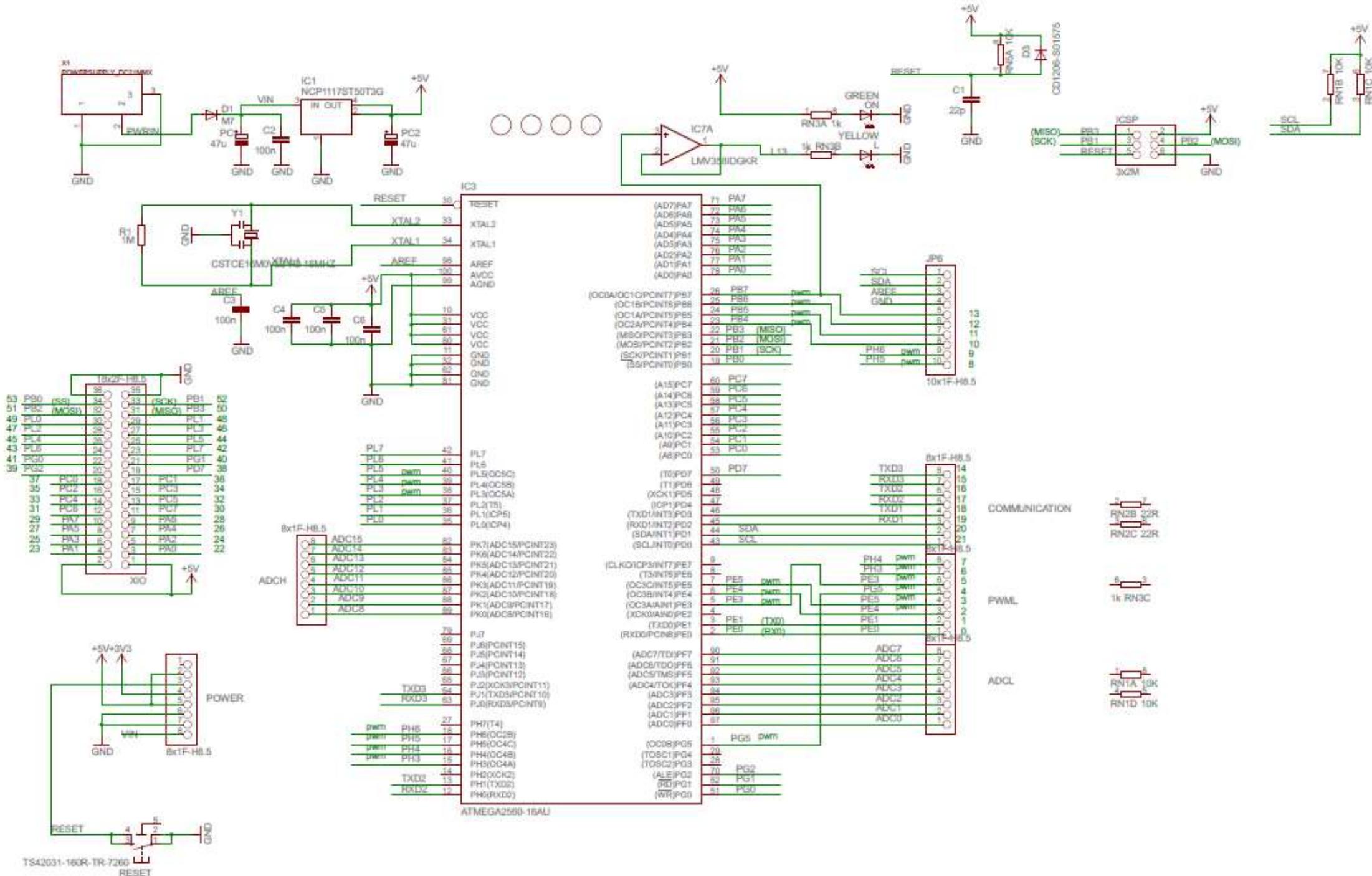
JTAG-Gravadores no circuito

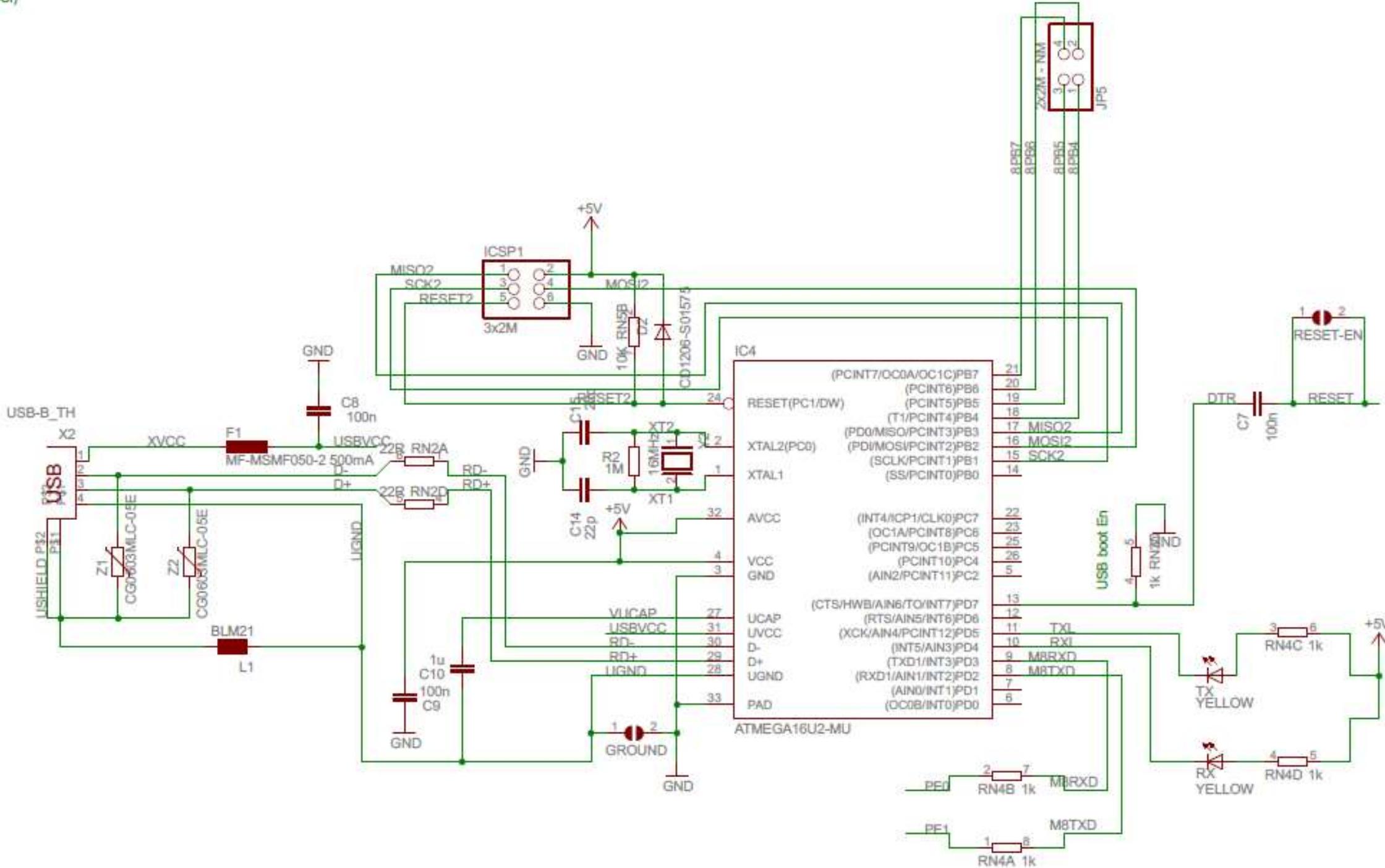


JTAG
PIC

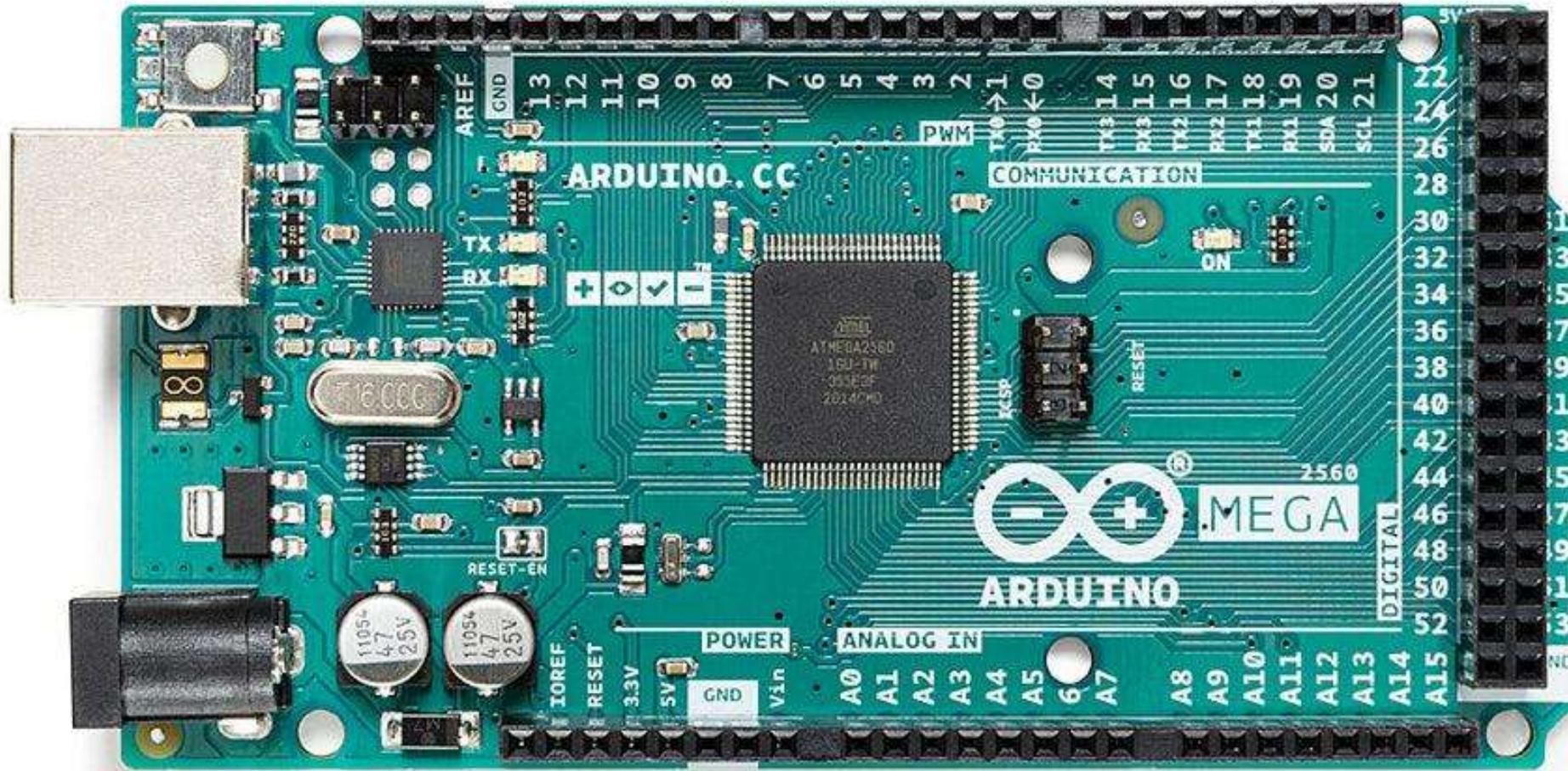


JTAG
Freescale





Arduino Mega



Microcontrolador Atmel2560

Sistemas Microcontrolados

Datasheet



Atmel ATmega640/V-1280/V-1281/V-2560/V-2561/V

8-bit Atmel Microcontroller with 16/32/64KB In-System Programmable Flash

DATASHEET

Features

- High Performance, Low Power Atmel® AVR® 8-Bit Microcontroller
- Advanced RISC Architecture
 - 135 Powerful Instructions – Most Single Clock Cycle Execution
 - 32 x 8 General Purpose Working Registers
 - Fully Static Operation
 - Up to 16 MIPS Throughput at 16MHz
 - On-Chip 2-cycle Multiplier
- High Endurance Non-volatile Memory Segments
 - 64K/128K/256Kbytes of In-System Self-Programmable Flash
 - 4Kbytes EEPROM
 - 8Kbytes Internal SRAM
 - Write/Erase Cycles: 10,000 Flash/100,000 EEPROM
 - Data retention: 20 years at 85°C/ 100 years at 25°C
 - Optional Boot Code Section with Independent Lock Bits
 - In-System Programming by On-chip Boot Program
 - True Read-While-Write Operation
 - Programming Lock for Software Security
 - Endurance: Up to 64Kbytes Optional External Memory Space
- Atmel® QTouch™ Library support
 - Capacitive touch buttons, sliders and wheels
 - QTouch and QMatrix acquisition
 - Up to 64 sense channels
- JTAG (IEEE® std. 1149.1 compliant) Interface
 - Boundary-scan Capabilities According to the JTAG Standard
 - Extensive On-chip Debug Support
 - Programming of Flash, EEPROM, Fuses, and Lock Bits through the JTAG Interface
- Peripheral Features
 - Two 8-bit Timer/Counters with Separate Prescaler and Compare Mode
 - Four 16-bit Timer/Counter with Separate Prescaler, Compare- and Capture Mode
 - Real Time Counter with Separate Oscillator
 - Four 8-bit PWM Channels
 - Six/Twelve PWM Channels with Programmable Resolution from 2 to 16 Bits (ATmega128/2561, ATmega640/1280/2560)
 - Output Compare Modulator
 - 8/16-channel, 10-/8-bit ADC (ATmega1281/2561, ATmega540/1280/2560)
 - Two/Four Programmable Serial USART (ATmega1281/2561, ATmega640/1280/2560)
 - Master/Slave SPI Serial Interface
 - Byte Oriented 2-wire Serial Interface
 - Programmable Watchdog Timer with Separate On-chip Oscillator
 - On-chip Analog Comparators
 - Interrupt and Wake-up on Pin Change
- Special Microcontroller Features
 - Power-on Reset and Programmable Brown-out Detection
 - Internal Calibrated Oscillator
 - External and Internal Interrupt Sources
 - Six Sleep Modes: Idle, ADC Noise Reduction, Power-save, Power-down, Standby, and Extended Standby
- I/O and Packages
 - 54/64 Programmable I/O Lines (ATmega1281/2561, ATmega640/1280/2560)
 - 64-pin QFN/MLF, 64-lead TQFP (ATmega128/256)
 - 100-lead TQFP, 108-lead CQGA (ATmega640/1280/2560)
 - RoHS/6RoHS Green
- Temperature Range:
 - -40°C to 85°C Industrial
- Ultra-Low Power Consumption
 - Active Mode: 1MHz: 1.8V: 560µA
 - Power-down Mode: 0.1µA at 1.8V
- Speed Grade:
 - ATmega640/VATmega1280/VATmega1281:
 - 0 - 4MHz @ 1.8V - 5.5V, 0 - 8MHz @ 2.7V - 5.5V
 - ATmega2560/VATmega3261:
 - 0 - 2MHz @ 1.8V - 5.5V, 0 - 8MHz @ 2.7V - 5.5V
 - ATmega480/VATmega1280/VATmega1281:
 - 0 - 8MHz @ 2.7V - 5.5V, 0 - 16MHz @ 4.5V - 5.5V
 - ATmega2560/VATmega5261:
 - 0 - 16MHz @ 4.5V - 5.5V

Datasheet

Features

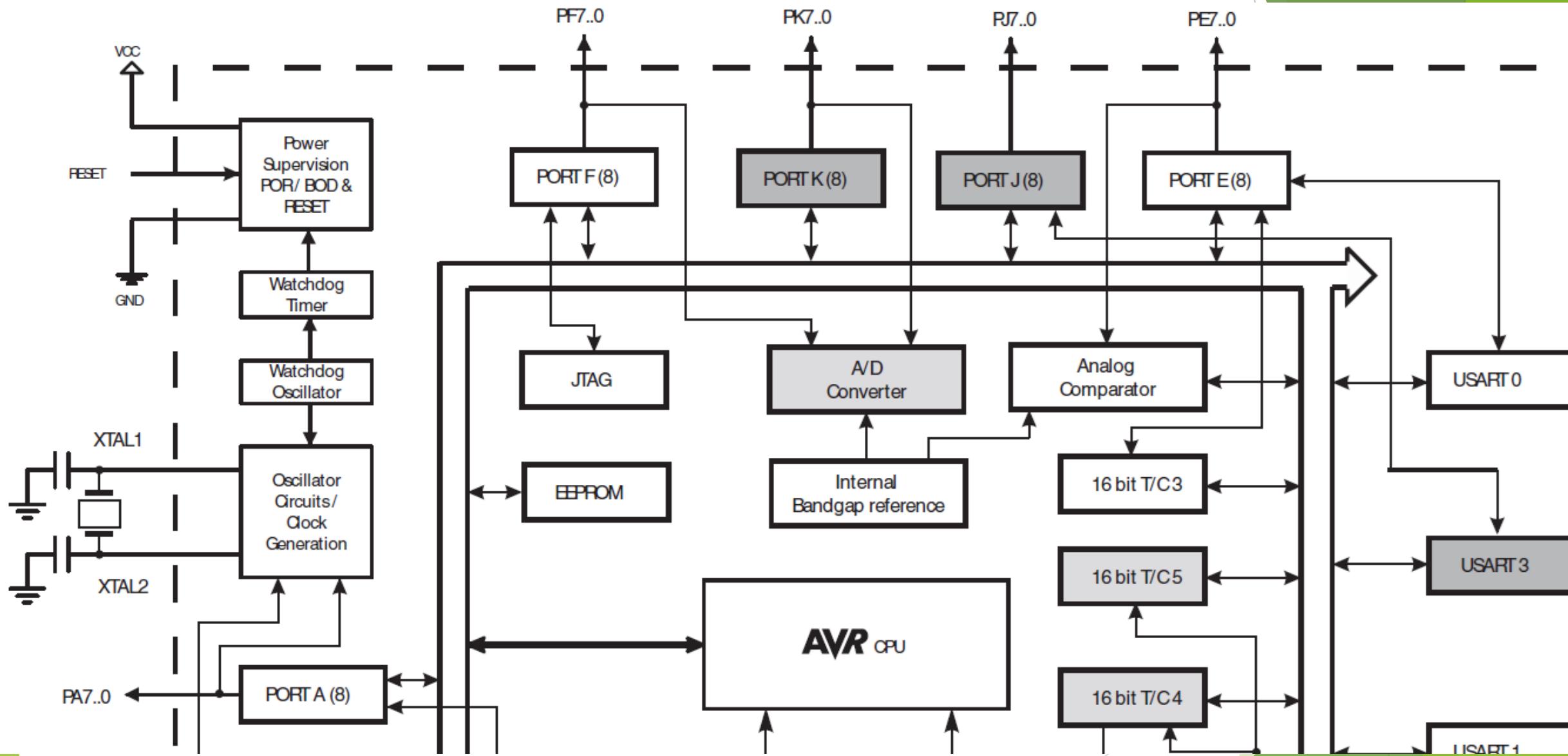
- High Performance, Low Power Atmel® AVR® 8-Bit Microcontroller
- Advanced RISC Architecture
 - 135 Powerful Instructions – Most Single Clock Cycle Execution
 - 32 × 8 General Purpose Working Registers
 - Fully Static Operation
 - Up to 16 MIPS Throughput at 16MHz
 - On-Chip 2-cycle Multiplier
- High Endurance Non-volatile Memory Segments
 - 64K/128K/256KBytes of In-System Self-Programmable Flash
 - 4Kbytes EEPROM
 - 8Kbytes Internal SRAM
 - Write/Erase Cycles:10,000 Flash/100,000 EEPROM
 - Data retention: 20 years at 85°C/ 100 years at 25°C
 - Optional Boot Code Section with Independent Lock Bits
 - In-System Programming by On-chip Boot Program
 - True Read-While-Write Operation
 - Programming Lock for Software Security
 - Endurance: Up to 64Kbytes Optional External Memory Space
- Atmel® QTouch® library support
 - Capacitive touch buttons, sliders and wheels
 - QTouch and QMatrix acquisition
 - Up to 64 sense channels
- JTAG (IEEE® std. 1149.1 compliant) Interface
 - Boundary-scan Capabilities According to the JTAG Standard
 - Extensive On-chip Debug Support
 - Programming of Flash, EEPROM, Fuses, and Lock Bits through the JTAG Interface

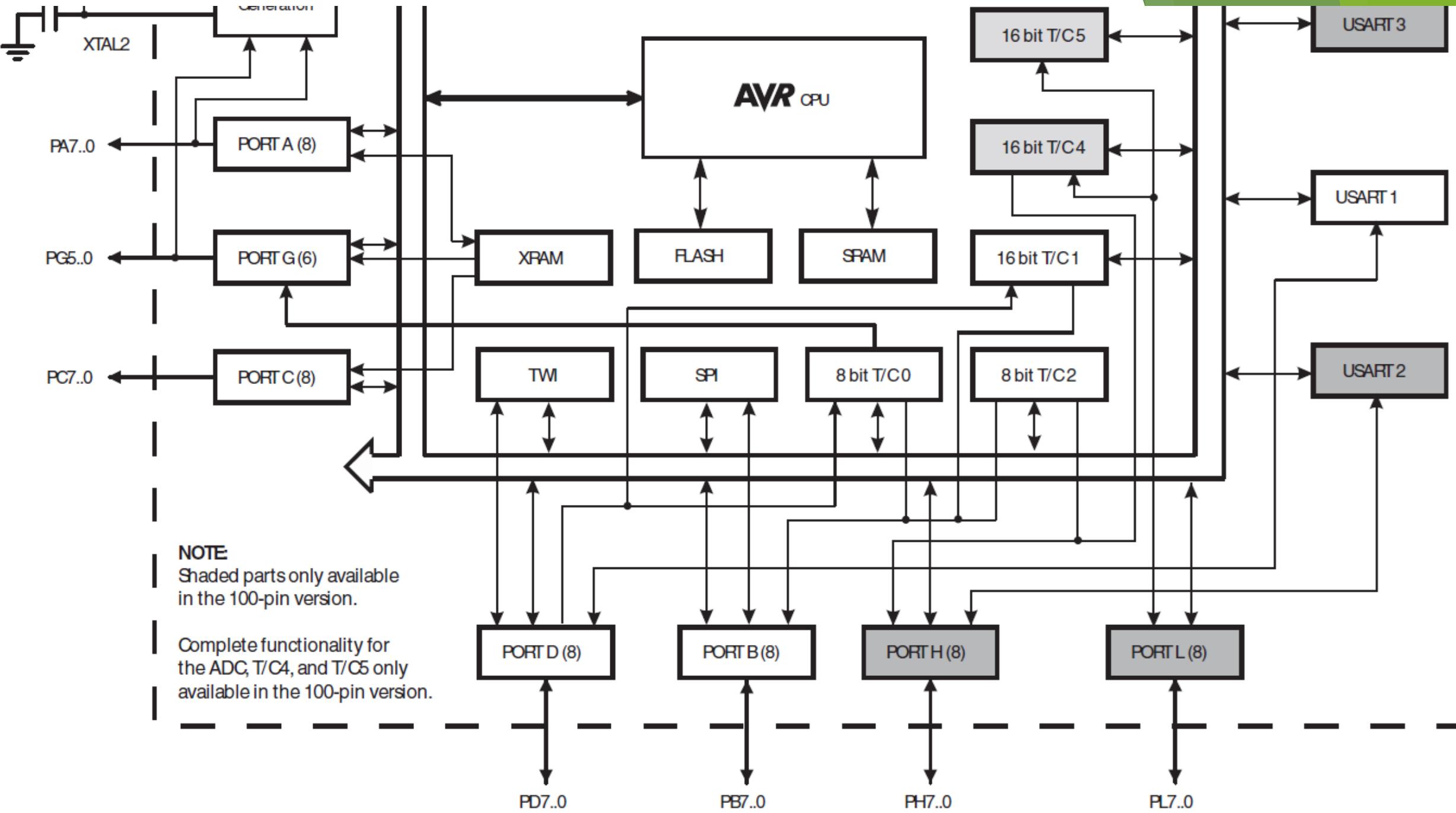
Datasheet

- Peripheral Features
 - Two 8-bit Timer/Counters with Separate Prescaler and Compare Mode
 - Four 16-bit Timer/Counter with Separate Prescaler, Compare- and Capture Mode
 - Real Time Counter with Separate Oscillator
 - Four 8-bit PWM |Channels
 - Six/Twelve PWM Channels with Programmable Resolution from 2 to 16 Bits (ATmega1281/2561, ATmega640/1280/2560)
 - Output Compare Modulator
 - 8/16-channel, 10-bit ADC (ATmega1281/2561, ATmega640/1280/2560)
 - Two/Four Programmable Serial USART (ATmega1281/2561, ATmega640/1280/2560)
 - Master/Slave SPI Serial Interface
 - Byte Oriented 2-wire Serial Interface
 - Programmable Watchdog Timer with Separate On-chip Oscillator
 - On-chip Analog Comparator
 - Interrupt and Wake-up on Pin Change
- Special Microcontroller Features
 - Power-on Reset and Programmable Brown-out Detection
 - Internal Calibrated Oscillator
 - External and Internal Interrupt Sources
 - Six Sleep Modes: Idle, ADC Noise Reduction, Power-save, Power-down, Standby, and Extended Standby
- I/O and Packages
 - 54/86 Programmable I/O Lines (ATmega1281/2561, ATmega640/1280/2560)
 - 64-pad QFN/MLF, 64-lead TQFP (ATmega1281/2561)
 - 100-lead TQFP, 100-ball CBGA (ATmega640/1280/2560)
 - RoHS/Fully Green

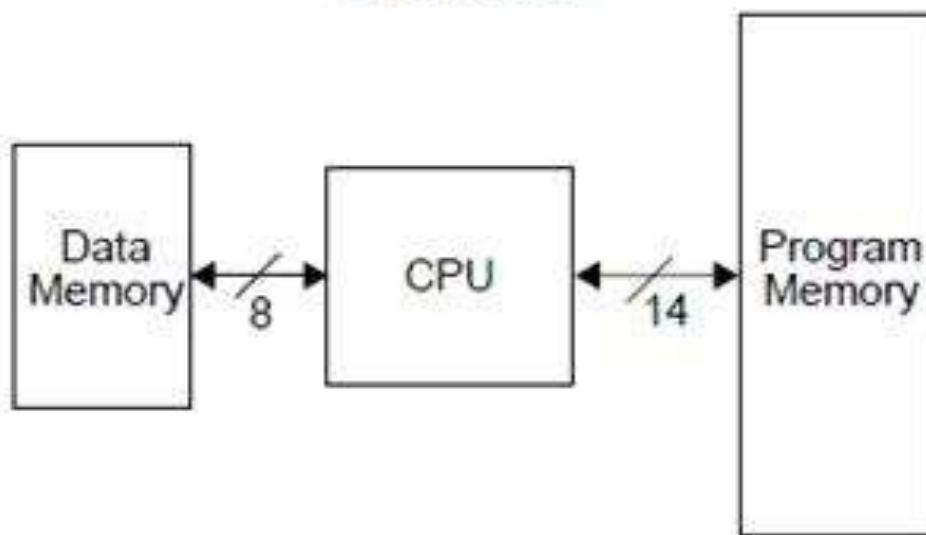
Datasheet

- **Temperature Range:**
 - -40°C to 85°C Industrial
- **Ultra-Low Power Consumption**
 - Active Mode: 1MHz, 1.8V: 500µA
 - Power-down Mode: 0.1µA at 1.8V
- **Speed Grade:**
 - ATmega640V/ATmega1280V/ATmega1281V:
 - 0 - 4MHz @ 1.8V - 5.5V, 0 - 8MHz @ 2.7V - 5.5V
 - ATmega2560V/ATmega2561V:
 - 0 - 2MHz @ 1.8V - 5.5V, 0 - 8MHz @ 2.7V - 5.5V
 - ATmega640/ATmega1280/ATmega1281:
 - 0 - 8MHz @ 2.7V - 5.5V, 0 - 16MHz @ 4.5V - 5.5V
 - ATmega2560/ATmega2561:
 - 0 - 16MHz @ 4.5V - 5.5V

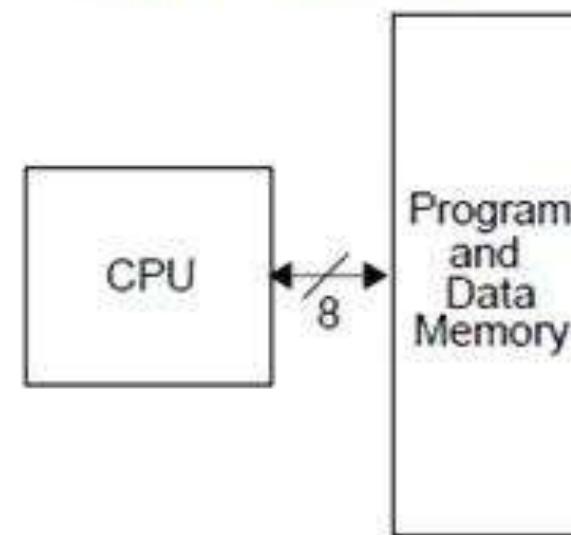




Harvard

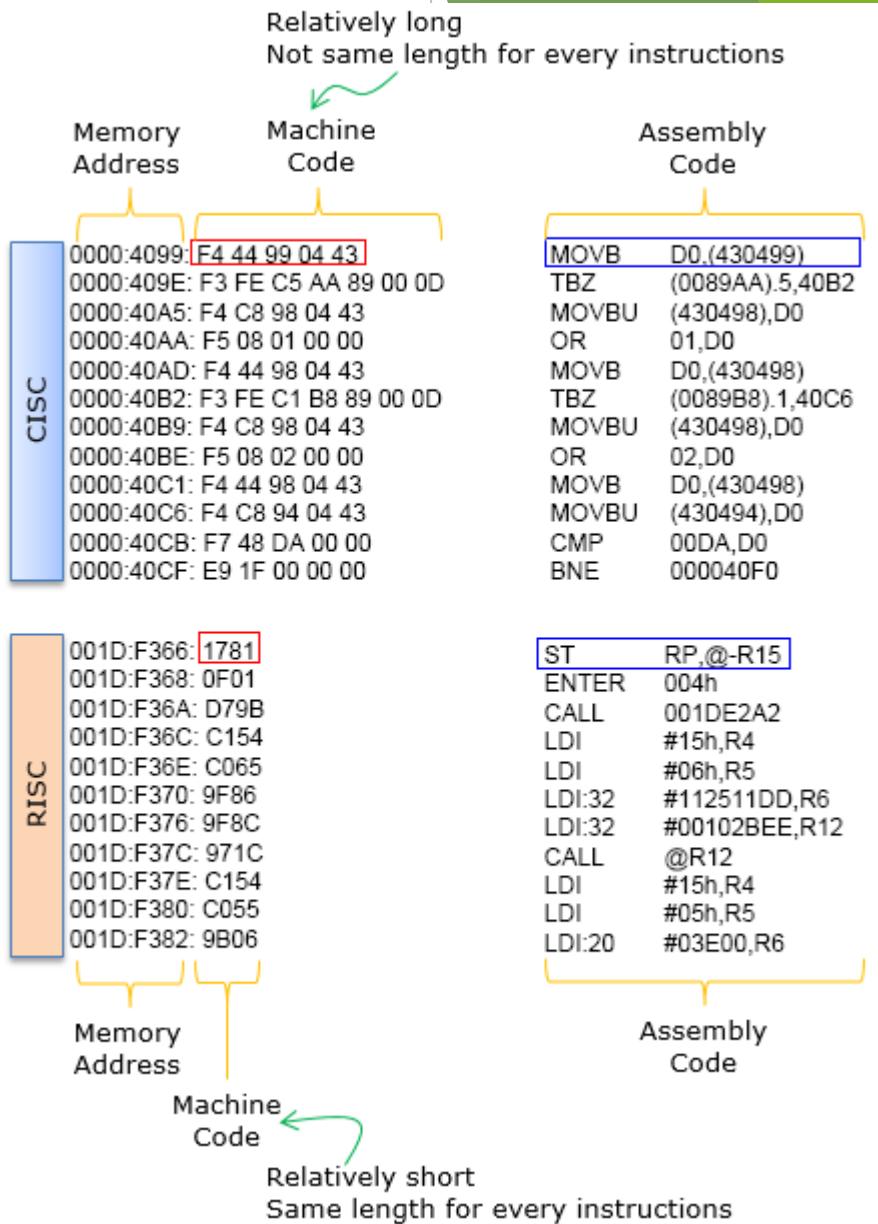


von-Neumann



CISC vs RISC

- ▶ CISC - Complex Instruction Set Computer
- ▶ RISC - Reduced Instruction Set Computer



Relatively long
Not same length for every instructions

Memory Address	Machine Code	Assembly Code
0000:4099:	F4 44 99 04 43	MOVB D0,(430499)
0000:409E:	F3 FE C5 AA 89 00 0D	TBZ (0089AA).5,40B2
0000:40A5:	F4 C8 98 04 43	MOVBU (430498),D0
0000:40AA:	F5 08 01 00 00	OR 01,D0
0000:40AD:	F4 44 98 04 43	MOVB D0,(430498)
0000:40B2:	F3 FE C1 B8 89 00 0D	TBZ (0089B8).1,40C6
0000:40B9:	F4 C8 98 04 43	MOVBU (430498),D0
0000:40BE:	F5 08 02 00 00	OR 02,D0
0000:40C1:	F4 44 98 04 43	MOVB D0,(430498)
0000:40C6:	F4 C8 94 04 43	MOVBU (430494),D0
0000:40CB:	F7 48 DA 00 00	CMP 00DA,D0
0000:40CF:	E9 1F 00 00 00	BNE 000040F0

RISC

```
001D:F366: 1781  
001D:F368: 0F01  
001D:F36A: D79B  
001D:F36C: C154  
001D:F36E: C065  
001D:F370: 9F86  
001D:F376: 9F8C  
001D:F37C: 971C  
001D:F37E: C154  
001D:F380: C055  
001D:F382: 9B06
```



Memory
Address

Machine
Code

Relatively short
Same length for every instructions

```
ST      RP,@-R15  
ENTER   004h  
CALL    001DE2A2  
LDI     #15h,R4  
LDI     #06h,R5  
LDI:32  #112511DD,R6  
LDI:32  #00102BEE,R12  
CALL    @R12  
LDI     #15h,R4  
LDI     #05h,R5  
LDI:20  #03E00,R6
```

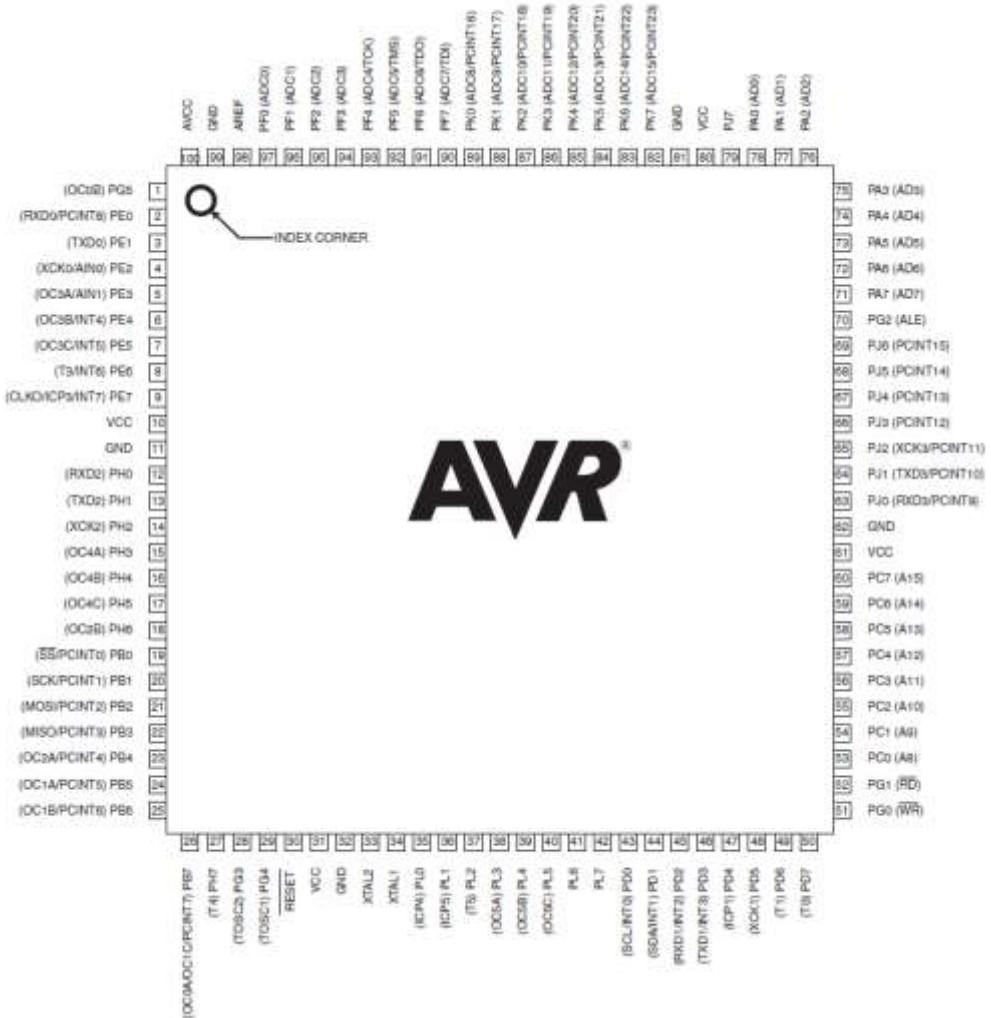
Assembly
Code

MIPS (*Millions of Instructions Per Second*)

Pinagem

Sistemas Microcontrolados

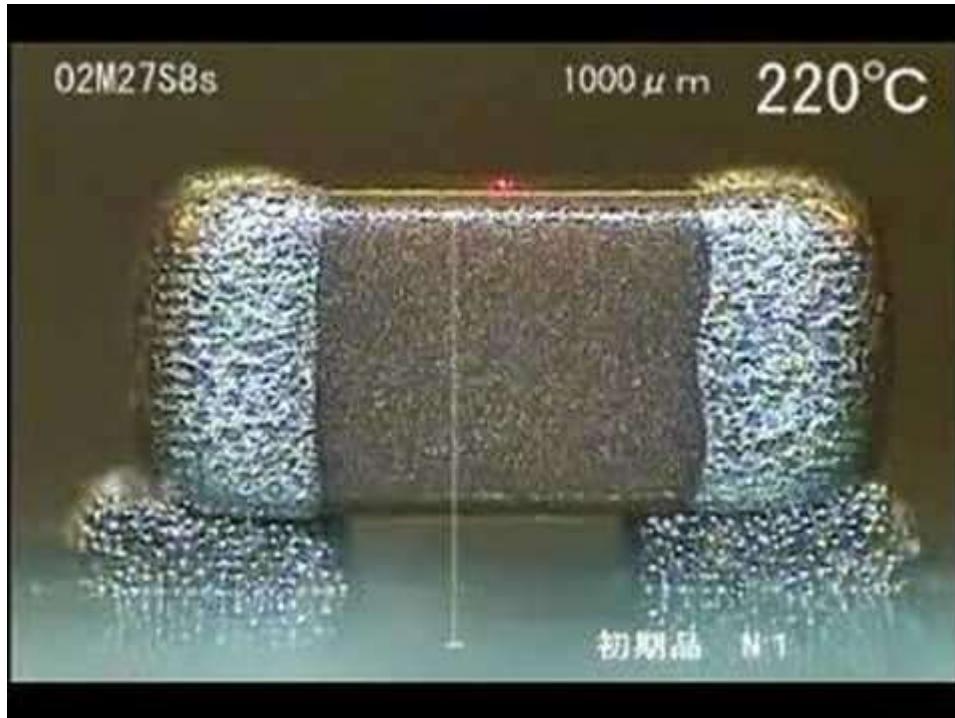
TQFP-pinout ATmega640/1280/2560



Quad Flat Package (QFP)
TQFP (*Thin QFP*)



Surface-mount technology (SMT)



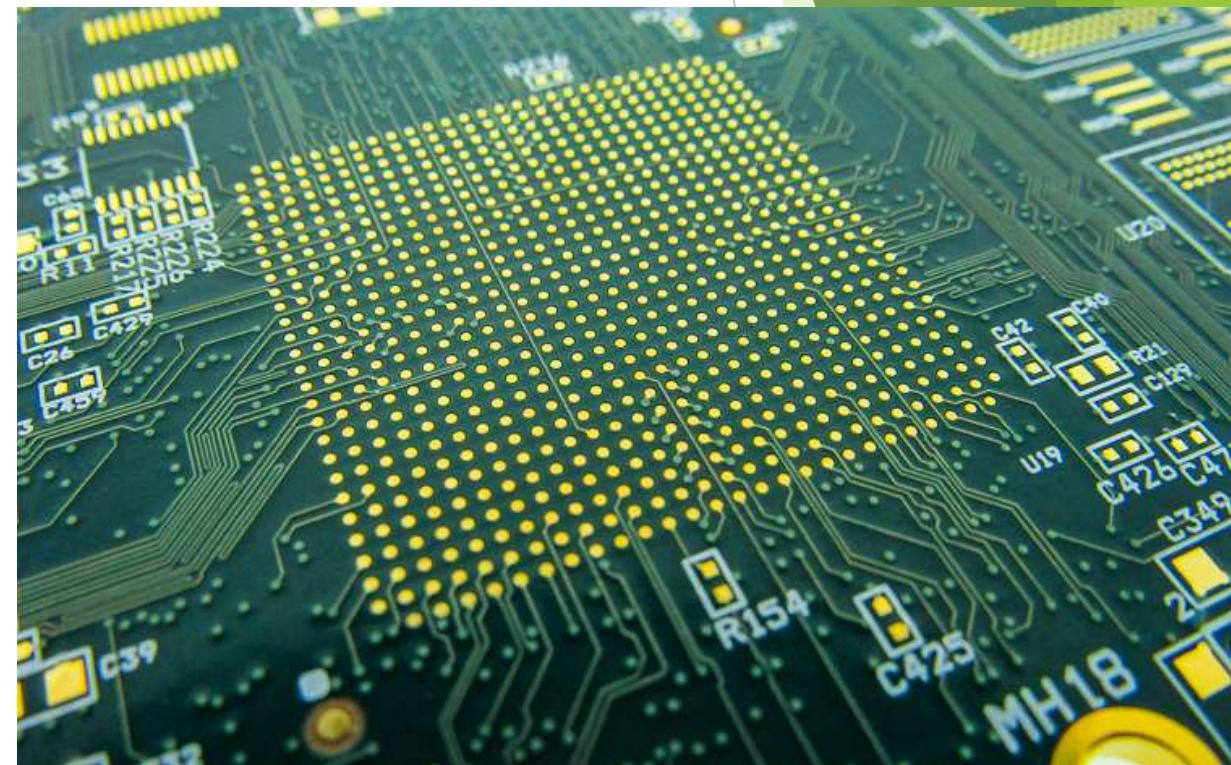
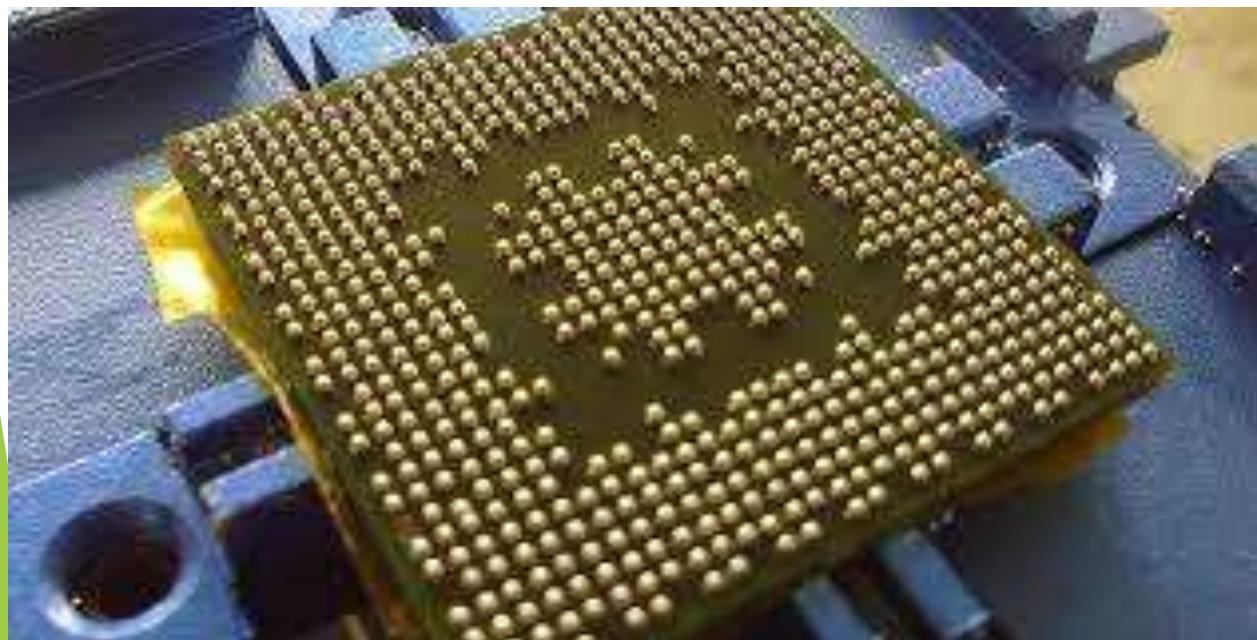
https://www.youtube.com/watch?v=o_SwxI2KTys



https://www.youtube.com/watch?v=S8qkaTsr2_o

<https://www.youtube.com/watch?v=tQHDSXDZYF4>

Solda BGA (ball grid array)



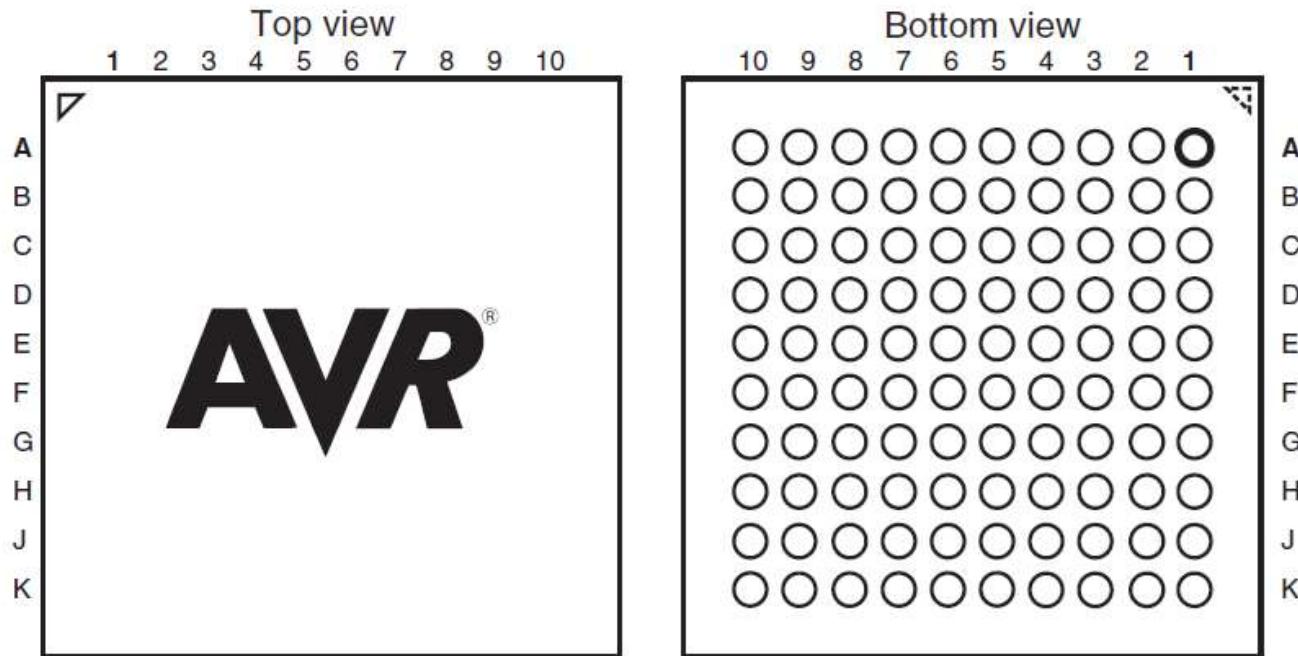
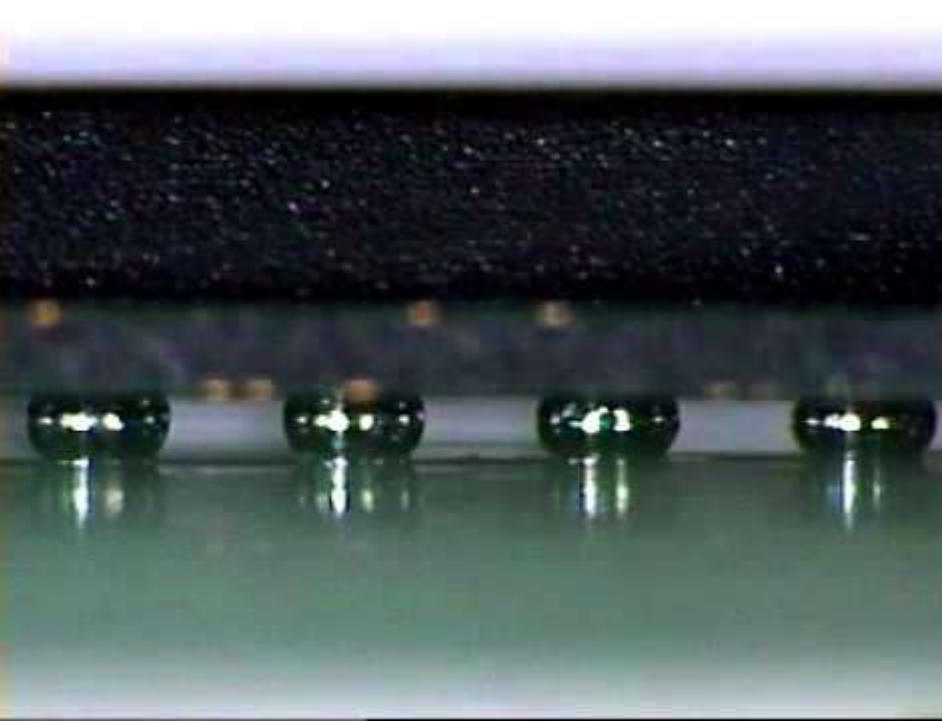


Table 1-1. CBGA-pinout ATmega640/1280/2560

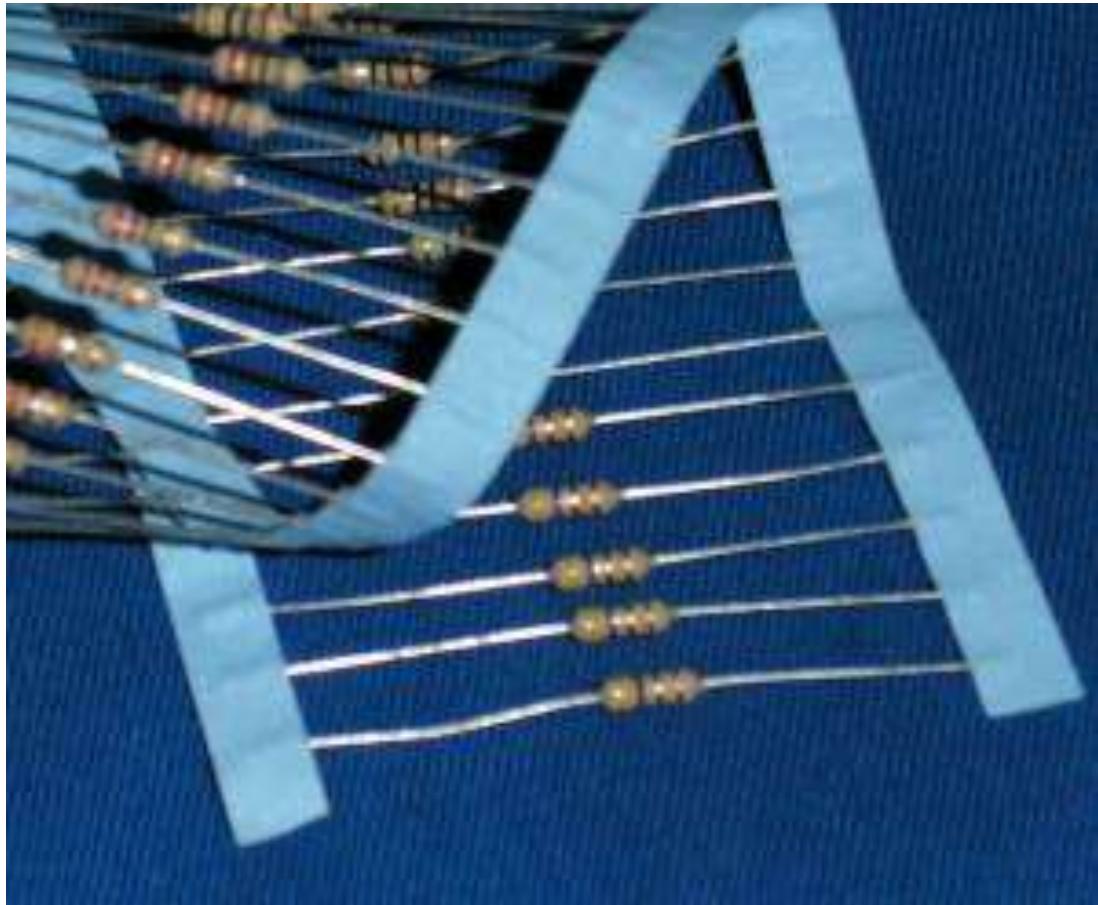
	1	2	3	4	5	6	7	8	9	10
A	GND	AREF	PF0	PF2	PF5	PK0	PK3	PK6	GND	VCC
B	AVCC	PG5	PF1	PF3	PF6	PK1	PK4	PK7	PA0	PA2
C	PE2	PE0	PE1	PF4	PF7	PK2	PK5	PJ7	PA1	PA3
D	PE3	PE4	PE5	PE6	PH2	PA4	PA5	PA6	PA7	PG2
E	PE7	PH0	PH1	PH3	PH5	PJ6	PJ5	PJ4	PJ3	PJ2
F	VCC	PH4	PH6	PB0	PL4	PD1	PJ1	PJ0	PC7	GND
G	GND	PB1	PB2	PB5	PL2	PD0	PD5	PC5	PC6	VCC
H	PB3	PB4	RESET	PL1	PL3	PL7	PD4	PC4	PC3	PC2
J	PH7	PG3	PB6	PL0	XTAL2	PL6	PD3	PC1	PC0	PG1
K	PB7	PG4	VCC	GND	XTAL1	PL5	PD2	PD6	PD7	PG0

Note: The functions for each pin is the same as for the 100 pin packages shown in [Figure 1-1 on page 2](#).



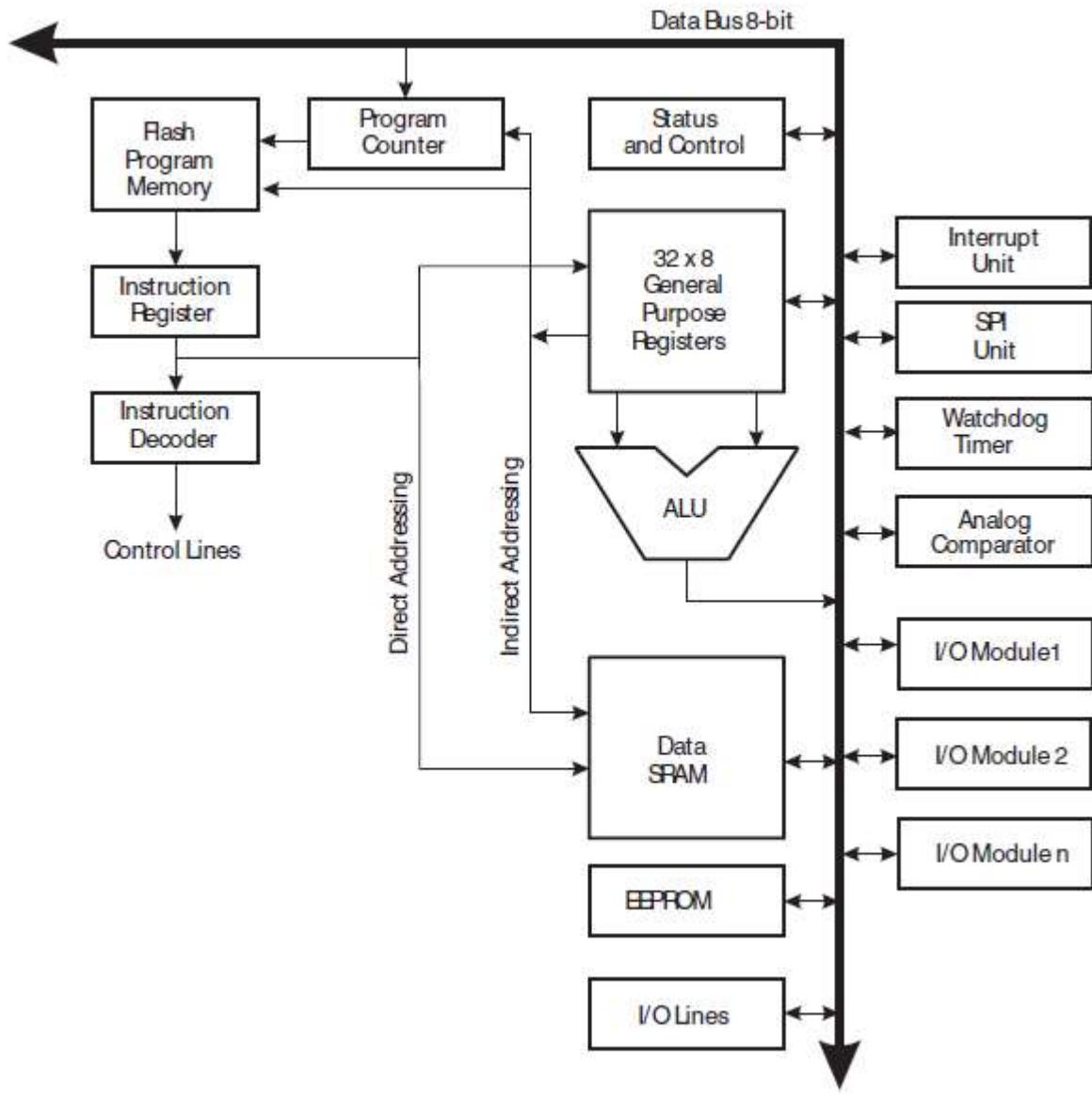
<https://www.youtube.com/watch?v=5mWHRakh5Hw>

Componente PTH (Plated through-hole)



Núcleo (AVR CPU Core)

Sistemas Microcontrolados



ALU - Arithmetic Logic Unit

Mnemonics	Operands	Description	Operation	Flags	#Clocks
ARITHMETIC AND LOGIC INSTRUCTIONS					
ADD	Rd, Rr	Add two Registers	$Rd \leftarrow Rd + Rr$	Z, C, N, V, H	1
ADC	Rd, Rr	Add with Carry two Registers	$Rd \leftarrow Rd + Rr + C$	Z, C, N, V, H	1
ADIW	Rdl,K	Add Immediate to Word	$Rdh:Rdl \leftarrow Rdh:Rdl + K$	Z, C, N, V, S	2
SUB	Rd, Rr	Subtract two Registers	$Rd \leftarrow Rd - Rr$	Z, C, N, V, H	1
SUBI	Rd, K	Subtract Constant from Register	$Rd \leftarrow Rd - K$	Z, C, N, V, H	1
SBC	Rd, Rr	Subtract with Carry two Registers	$Rd \leftarrow Rd - Rr - C$	Z, C, N, V, H	1
SBCI	Rd, K	Subtract with Carry Constant from Reg.	$Rd \leftarrow Rd - K - C$	Z, C, N, V, H	1
SBIW	Rdl,K	Subtract Immediate from Word	$Rdh:Rdl \leftarrow Rdh:Rdl - K$	Z, C, N, V, S	2
AND	Rd, Rr	Logical AND Registers	$Rd \leftarrow Rd \bullet Rr$	Z, N, V	1
ANDI	Rd, K	Logical AND Register and Constant	$Rd \leftarrow Rd \bullet K$	Z, N, V	1
OR	Rd, Rr	Logical OR Registers	$Rd \leftarrow Rd \vee Rr$	Z, N, V	1
ORI	Rd, K	Logical OR Register and Constant	$Rd \leftarrow Rd \vee K$	Z, N, V	1
EOR	Rd, Rr	Exclusive OR Registers	$Rd \leftarrow Rd \oplus Rr$	Z, N, V	1
COM	Rd	One's Complement	$Rd \leftarrow 0xFF - Rd$	Z, C, N, V	1
NEG	Rd	Two's Complement	$Rd \leftarrow 0x00 - Rd$	Z, C, N, V, H	1
SBR	Rd,K	Set Bit(s) in Register	$Rd \leftarrow Rd \vee K$	Z, N, V	1
CBR	Rd,K	Clear Bit(s) in Register	$Rd \leftarrow Rd \bullet (0xFF - K)$	Z, N, V	1
INC	Rd	Increment	$Rd \leftarrow Rd + 1$	Z, N, V	1
DEC	Rd	Decrement	$Rd \leftarrow Rd - 1$	Z, N, V	1
TST	Rd	Test for Zero or Minus	$Rd \leftarrow Rd \bullet Rd$	Z, N, V	1

ALU - Arithmetic Logic Unit

		FUNCTION	OPERATION	FLAGS	CYCLES
CLR	Rd	Clear Register	$Rd \leftarrow Rd \oplus Rd$	Z, N, V	1
SER	Rd	Set Register	$Rd \leftarrow 0xFF$	None	1
MUL	Rd, Rr	Multiply Unsigned	$R1:R0 \leftarrow Rd \times Rr$	Z, C	2
MULS	Rd, Rr	Multiply Signed	$R1:R0 \leftarrow Rd \times Rr$	Z, C	2
MULSU	Rd, Rr	Multiply Signed with Unsigned	$R1:R0 \leftarrow Rd \times Rr$	Z, C	2
FMUL	Rd, Rr	Fractional Multiply Unsigned	$R1:R0 \leftarrow (Rd \times Rr) \lll 1$	Z, C	2
FMULS	Rd, Rr	Fractional Multiply Signed	$R1:R0 \leftarrow (Rd \times Rr) \lll 1$	Z, C	2
FMULSU	Rd, Rr	Fractional Multiply Signed with Unsigned	$R1:R0 \leftarrow (Rd \times Rr) \lll 1$	Z, C	2
BRANCH INSTRUCTIONS					
RJMP	k	Relative Jump	$PC \leftarrow PC + k + 1$	None	2
IJMP		Indirect Jump to (Z)	$PC \leftarrow Z$	None	2
EIJMP		Extended Indirect Jump to (Z)	$PC \leftarrow (EIND:Z)$	None	2
JMP	k	Direct Jump	$PC \leftarrow k$	None	3
RCALL	k	Relative Subroutine Call	$PC \leftarrow PC + k + 1$	None	4
ICALL		Indirect Call to (Z)	$PC \leftarrow Z$	None	4
EICALL		Extended Indirect Call to (Z)	$PC \leftarrow (EIND:Z)$	None	4
CALL	k	Direct Subroutine Call	$PC \leftarrow k$	None	5
RET		Subroutine Return	$PC \leftarrow STACK$	None	5
RETI		Interrupt Return	$PC \leftarrow STACK$	I	5
CPSE	Rd,Rr	Compare, Skip if Equal	if $(Rd = Rr)$ $PC \leftarrow PC + 2$ or 3	None	1/2/3
CP	Rd,Rr	Compare	$Rd - Rr$	Z, N, V, C, H	1
CPC	Rd,Rr	Compare with Carry	$Rd - Rr - C$	Z, N, V, C, H	1

Status Register

- ▶ O *Status Register* contém informações sobre os resultados de operações aritméticas mais recentes.
- ▶ Estas informações podem ser usadas para alterar o fluxo de programa para atender instruções condicionais.
- ▶ Portanto o *Status Register* é atualizado após cada operação da ALU
- ▶ Em muitos casos exclui-se a necessidade de utilizar instruções de comparação dedicadas resultando em maior velocidade e código mais compacto.

7.4.1 SREG – AVR Status Register

The AVR Status Register – SREG – is defined as:

Bit	7	6	5	4	3	2	1	0	
0x3F (0x5F)	I	T	H	S	V	N	Z	C	SREG
Read/Write	R/W								
Initial Value	0	0	0	0	0	0	0	0	

- **Bit 7 – I: Global Interrupt Enable**

The Global Interrupt Enable bit must be set for the interrupts to be enabled. The individual interrupt enable control is then performed in separate control registers. If the Global Interrupt Enable Register is cleared, none of the interrupts are enabled independent of the individual interrupt enable settings. The I-bit is cleared by hardware after an interrupt has occurred, and is set by the RETI instruction to enable subsequent interrupts. The I-bit can also be set and cleared by the application with the SEI and CLI instructions, as described in the “[Instruction Set Summary](#)” on [page 404](#).

- **Bit 6 – T: Bit Copy Storage**

The Bit Copy instructions BLD (Bit LoaD) and BST (Bit STore) use the T-bit as source or destination for the operated bit. A bit from a register in the Register File can be copied into T by the BST instruction, and a bit in T can be copied into a bit in a register in the Register File by the BLD instruction.

BST	Rr, b	Bit Store from Register to T	$T \leftarrow Rr(b)$	T	1
BLD	Rd, b	Bit load from T to Register	$Rd(b) \leftarrow T$	None	1

- **Bit 5 – H: Half Carry Flag**

The Half Carry Flag H indicates a Half Carry in some arithmetic operations. Half Carry Is useful in BCD arithmetic. See the “[Instruction Set Summary](#)” on page 404 for detailed information.

- **Bit 4 – S: Sign Bit, $S = N \oplus V$**

The S-bit is always an exclusive or between the Negative Flag N and the Two's Complement Overflow Flag V. See the “[Instruction Set Summary](#)” on page 404 for detailed information.

- **Bit 3 – V: Two's Complement Overflow Flag**

The Two's Complement Overflow Flag V supports two's complement arithmetics. See the “[Instruction Set Summary](#)” on page 404 for detailed information.

- **Bit 2 – N: Negative Flag**

The Negative Flag N indicates a negative result in an arithmetic or logic operation. See the “[Instruction Set Summary](#)” on page 404 for detailed information.

- **Bit 1 – Z: Zero Flag**

The Zero Flag Z indicates a zero result in an arithmetic or logic operation. See the “[Instruction Set Summary](#)” on page 404 for detailed information.

- **Bit 0 – C: Carry Flag**

The Carry Flag C indicates a carry in an arithmetic or logic operation. See the “[Instruction Set Summary](#)” on page 404 for detailed information.

General Purpose Register File

Figure 7-2. AVR CPU General Purpose Working Registers

General Purpose Working Registers	7	0	Addr.	
	R0		0x00	
	R1		0x01	
	R2		0x02	
	...			
	R13		0x0D	
	R14		0x0E	
	R15		0x0F	
	R16		0x10	
	R17		0x11	
	...			
	R26		0x1A	X-register Low Byte
	R27		0x1B	X-register High Byte
	R28		0x1C	Y-register Low Byte
	R29		0x1D	Y-register High Byte
	R30		0x1E	Z-register Low Byte
	R31		0x1F	Z-register High Byte

Stack Pointer

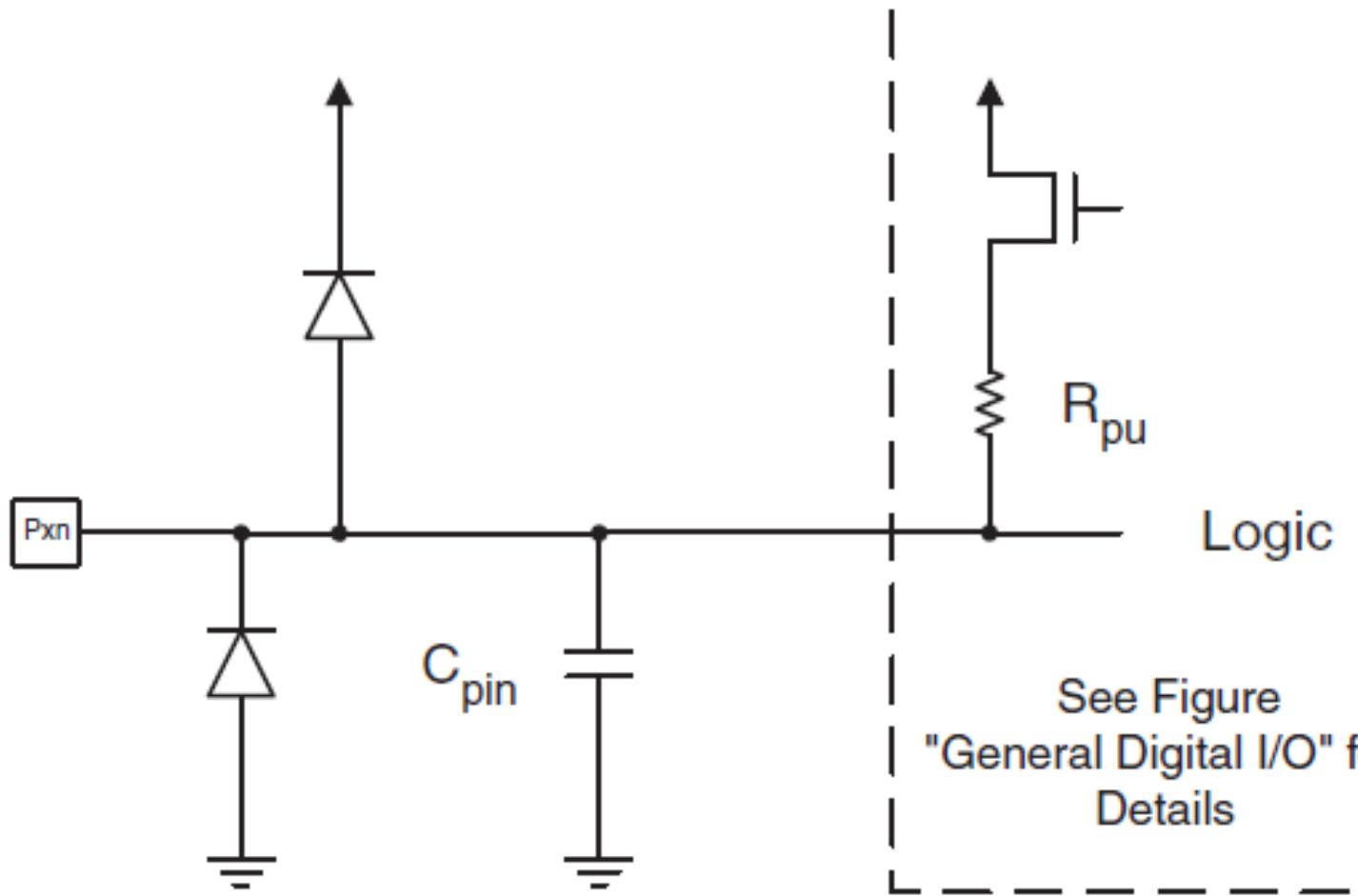
- ▶ A Pilha “Stack” é usado principalmente para armazenar dados temporariamente, variáveis locais e endereço de retorno após interrupções e chamadas de sub-rotinas.
- ▶ O registrador de ponteiro de pilha “Stack Pointer Register” sempre aponta para o topo da pilha.
- ▶ Note que a pilha é implementada de forma decrescente, de posições de memória maiores para menores. Isso significa que o comando “Stack PUSH” decrementa o ponteiro de memória.

Stack Pointer

- ▶ O stack pointer deve estar configurado para o endereço 0x0200.
- ▶ O endereço inicial da Pilha é o ultimo endereço da SRAM interna.
- ▶ O Stack pointer é decrementado quando dados são empurrados na pilha pela instrução PUSH e incrementado quando os dados forem retirado com a instrução POP.

I/O Ports (Portas de entrada e saída)

Sistemas Microcontrolados



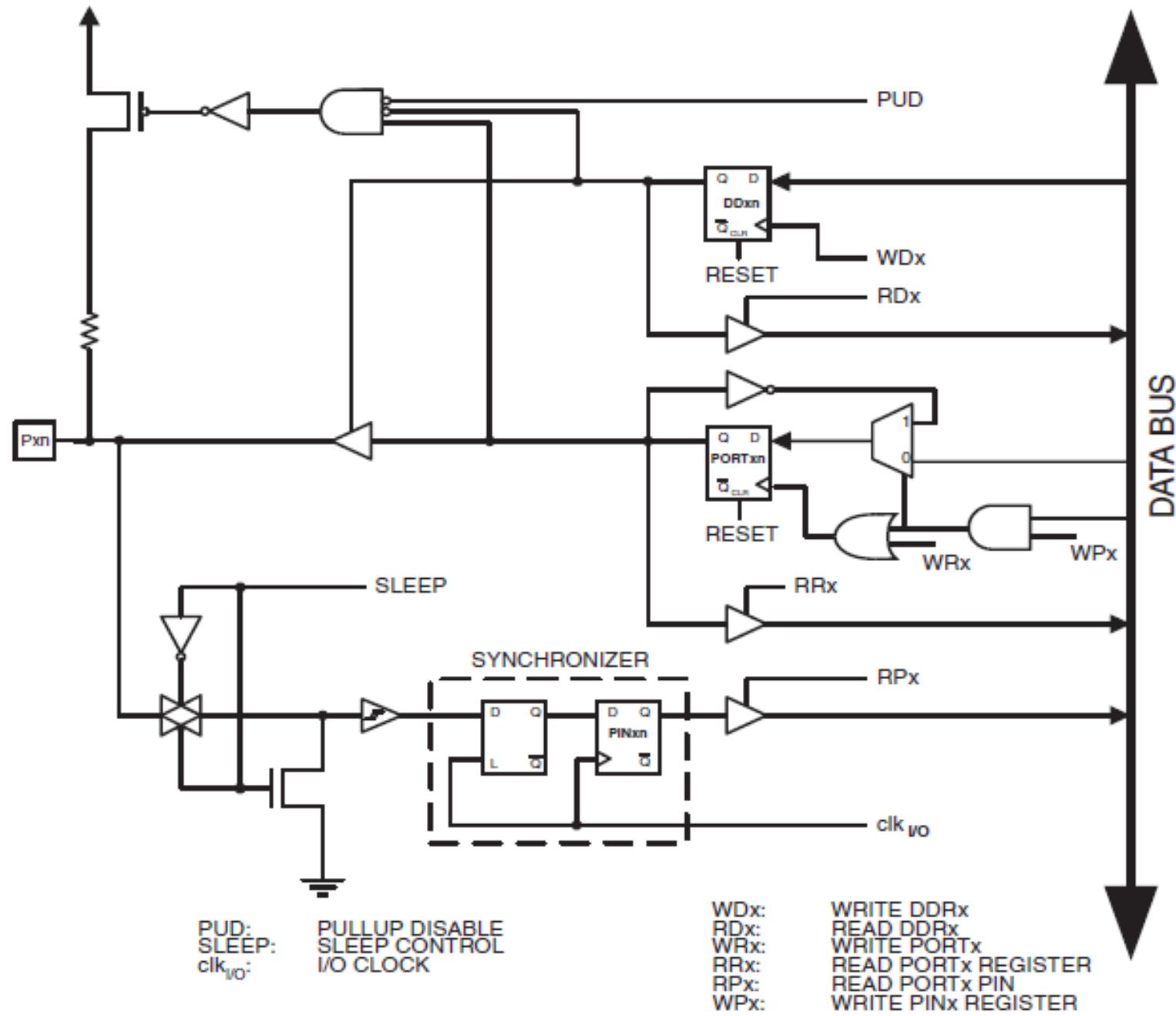
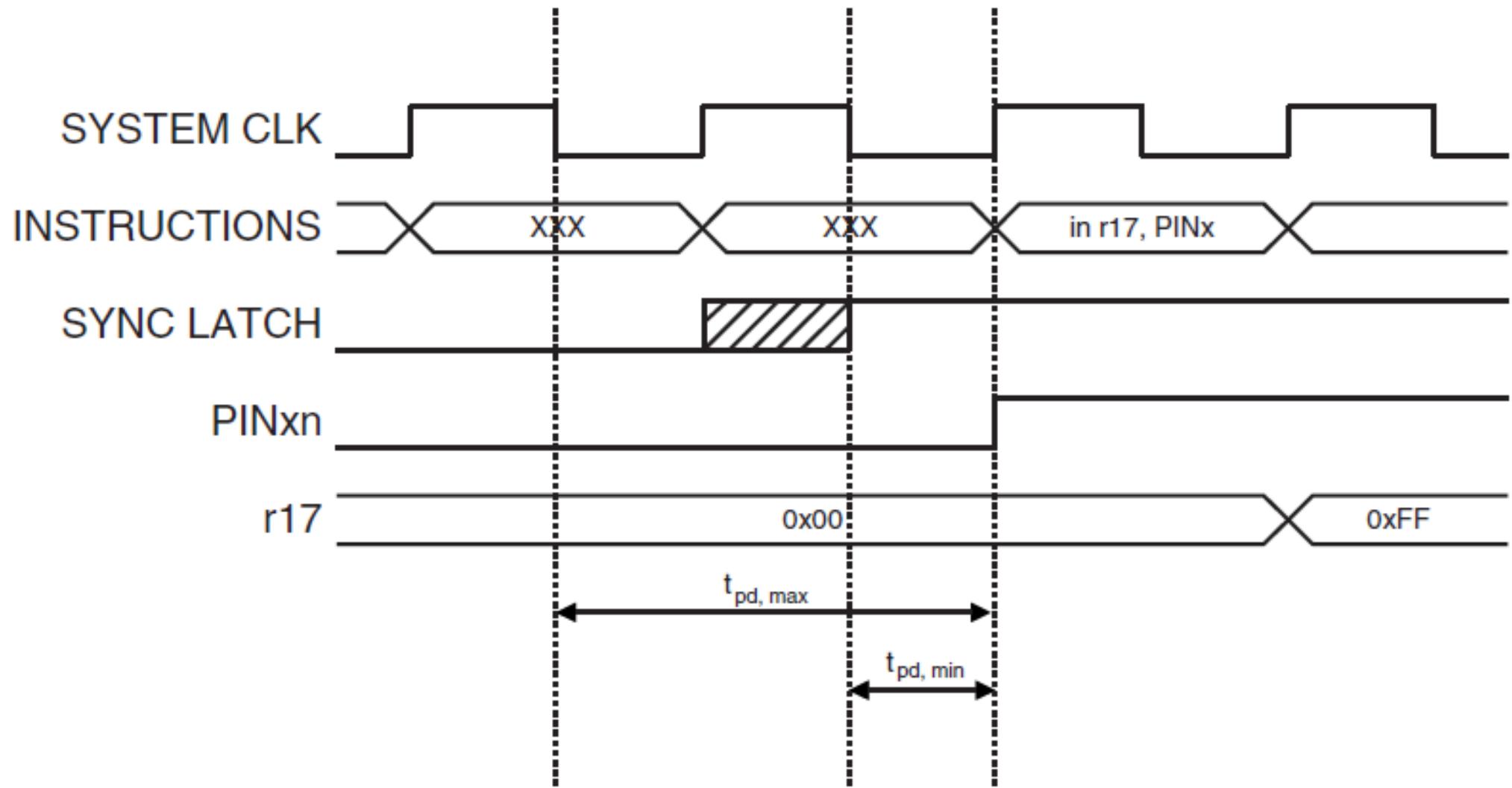


Figure 13-3. Synchronization when Reading an Externally Applied Pin value



Assembly Code Example⁽¹⁾

```
...
; Define pull-ups and set outputs high
; Define directions for port pins
ldi r16, (1<<PB7) | (1<<PB6) | (1<<PB1) | (1<<PB0)
ldi r17, (1<<DDB3) | (1<<DDB2) | (1<<DDB1) | (1<<DDB0)
out PORTB, r16
out DDRB, r17
; Insert nop for synchronization
nop
; Read port pins
in r16, PINB
...
```

C Code Example

```
unsigned char i;  
...  
/* Define pull-ups and set outputs high */  
/* Define directions for port pins */  
PORTB = (1<<PB7) | (1<<PB6) | (1<<PB1) | (1<<PB0);  
DDRB = (1<<DDB3) | (1<<DDB2) | (1<<DDB1) | (1<<DDB0);  
/* Insert nop for synchronization*/  
_no_operation();  
/* Read port pins */  
i = PINB;  
...  
...
```

DDxn	PORTxn	PUD (in MCUCR)	I/O	Pull-up	Comment
0	0	X	Input	No	Tri-state (Hi-Z)
0	1	0	Input	Yes	Pxn will source current if ext. pulled low
0	1	1	Input	No	Tri-state (Hi-Z)
1	0	X	Output	No	Output Low (Sink)
1	1	X	Output	No	Output High (Source)

13.4.1 MCUCR – MCU Control Register

Bit	7	6	5	4	3	2	1	0	
0x35 (0x55)	JTD	-	-	PUD	-	-	IVSEL	IVCE	MCUCR
Read/Write	R/W	R	R	R/W	R	R	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

- **Bit 4 – PUD: Pull-up Disable**

When this bit is written to one, the I/O ports pull-up resistors are disabled even if the DDxn and PORTxn Registers are configured to enable the pull-up resistor ($\{DDxn, PORTxn\} = 0b01$). See “Configuring the Pin” on page 68 for more details about this feature.

13.4.2 PORTA – Port A Data Register

13.4.3 DDRA – Port A Data Direction Register

13.4.4 PINA – Port A Input Pins Address

13.3.1 Alternate Functions of Port A

The Port A has an alternate function as the address low byte and data lines for the External Memory Interface.

Table 13-3. Port A Pins Alternate Functions

Port Pin	Alternate Function
PA7	AD7 (External memory interface address and data bit 7)
PA6	AD6 (External memory interface address and data bit 6)
PA5	AD5 (External memory interface address and data bit 5)
PA4	AD4 (External memory interface address and data bit 4)
PA3	AD3 (External memory interface address and data bit 3)
PA2	AD2 (External memory interface address and data bit 2)
PA1	AD1 (External memory interface address and data bit 1)
PA0	AD0 (External memory interface address and data bit 0)

13.3.2 Alternate Functions of Port B

The Port B pins with alternate functions are shown in [Table 13-6](#).

Table 13-6. Port B Pins Alternate Functions

Port Pin	Alternate Functions
PB7	OC0A/OC1C/PCINT7 (Output Compare and PWM Output A for Timer/Counter0, Output Compare and PWM Output C for Timer/Counter1 or Pin Change Interrupt 7)
PB6	OC1B/PCINT6 (Output Compare and PWM Output B for Timer/Counter1 or Pin Change Interrupt 6)
PB5	OC1A/PCINT5 (Output Compare and PWM Output A for Timer/Counter1 or Pin Change Interrupt 5)
PB4	OC2A/PCINT4 (Output Compare and PWM Output A for Timer/Counter2 or Pin Change Interrupt 4)
PB3	MISO/PCINT3 (SPI Bus Master Input/Slave Output or Pin Change Interrupt 3)
PB2	MOSI/PCINT2 (SPI Bus Master Output/Slave Input or Pin Change Interrupt 2)
PB1	SCK/PCINT1 (SPI Bus Serial Clock or Pin Change Interrupt 1)
PB0	SS/PCINT0 (SPI Slave Select input or Pin Change Interrupt 0)

13.3.3 Alternate Functions of Port C

The Port C alternate function is as follows:

Table 13-9. Port C Pins Alternate Functions

Port Pin	Alternate Function
PC7	A15 (External Memory interface address bit 15)
PC6	A14 (External Memory interface address bit 14)
PC5	A13 (External Memory interface address bit 13)
PC4	A12 (External Memory interface address bit 12)
PC3	A11 (External Memory interface address bit 11)
PC2	A10 (External Memory interface address bit 10)
PC1	A9 (External Memory interface address bit 9)
PC0	A8 (External Memory interface address bit 8)

13.3.5 Alternate Functions of Port E

The Port E pins with alternate functions are shown in [Table 13-15](#).

Table 13-15. Port E Pins Alternate Functions

Port Pin	Alternate Function
PE7	INT7/ICP3/CLK0 (External Interrupt 7 Input, Timer/Counter3 Input Capture Trigger or Divided System Clock)
PE6	INT6/ T3 (External Interrupt 6 Input or Timer/Counter3 Clock Input)
PE5	INT5/OC3C (External Interrupt 5 Input or Output Compare and PWM Output C for Timer/Counter3)
PE4	INT4/OC3B (External Interrupt4 Input or Output Compare and PWM Output B for Timer/Counter3)
PE3	AIN1/OC3A (Analog Comparator Negative Input or Output Compare and PWM Output A for Timer/Counter3)
PE2	AIN0/XCK0 (Analog Comparator Positive Input or USART0 external clock input/output)
PE1	PDO ⁽¹⁾ /TXD0 (Programming Data Output or USART0 Transmit Pin)
PE0	PDI ⁽¹⁾ /RXD0/PCINT8 (Programming Data Input, USART0 Receive Pin or Pin Change Interrupt 8)

13.3.6 Alternate Functions of Port F

The Port F has an alternate function as analog input for the ADC as shown in [Table 13-18](#). If some Port F pins are configured as outputs, it is essential that these do not switch when a conversion is in progress. This might corrupt the result of the conversion. If the JTAG interface is enabled, the pull-up resistors on pins PF7(TDI), PF5(TMS), and PF4(TCK) will be activated even if a Reset occurs.

Table 13-18. Port F Pins Alternate Functions

Port Pin	Alternate Function
PF7	ADC7/TDI (ADC input channel 7 or JTAG Test Data Input)
PF6	ADC6/TDO (ADC input channel 6 or JTAG Test Data Output)
PF5	ADC5/TMS (ADC input channel 5 or JTAG Test Mode Select)
PF4	ADC4/TCK (ADC input channel 4 or JTAG Test Clock)
PF3	ADC3 (ADC input channel 3)
PF2	ADC2 (ADC input channel 2)
PF1	ADC1 (ADC input channel 1)
PF0	ADC0 (ADC input channel 0)

13.3.7 Alternate Functions of Port G

The Port G alternate pin configuration is as follows:

Table 13-21. Port G Pins Alternate Functions

Port Pin	Alternate Function
PG5	OC0B (Output Compare and PWM Output B for Timer/Counter0)
PG4	TOSC1 (RTC Oscillator Timer/Counter2)
PG3	TOSC2 (RTC Oscillator Timer/Counter2)
PG2	ALE (Address Latch Enable to external memory)
PG1	\overline{RD} (Read strobe to external memory)
PG0	\overline{WR} (Write strobe to external memory)

13.3.8 Alternate Functions of Port H

The Port H alternate pin configuration is as follows:

Table 13-24. Port H Pins Alternate Functions

Port Pin	Alternate Function
PH7	T4 (Timer/Counter4 Clock Input)
PH6	OC2B (Output Compare and PWM Output B for Timer/Counter2)
PH5	OC4C (Output Compare and PWM Output C for Timer/Counter4)
PH4	OC4B (Output Compare and PWM Output B for Timer/Counter4)
PH3	OC4A (Output Compare and PWM Output A for Timer/Counter4)
PH2	XCK2 (USART2 External Clock)
PH1	TXD2 (USART2 Transmit Pin)
PH0	RXD2 (USART2 Receive Pin)

13.3.9

Alternate Functions of Port J

The Port J alternate pin configuration is as follows:

Table 13-27. Port J Pins Alternate Functions

Port Pin	Alternate Function
PJ7	—
PJ6	PCINT15 (Pin Change Interrupt 15)
PJ5	PCINT14 (Pin Change Interrupt 14)
PJ4	PCINT13 (Pin Change Interrupt 13)
PJ3	PCINT12 (Pin Change Interrupt 12)
PJ2	XCK3/PCINT11 (USART3 External Clock or Pin Change Interrupt 11)
PJ1	TXD3/PCINT10 (USART3 Transmit Pin or Pin Change Interrupt 10)
PJ0	RXD3/PCINT9 (USART3 Receive Pin or Pin Change Interrupt 9)

13.3.10 Alternate Functions of Port K

The Port K alternate pin configuration is as follows:

Table 13-30. Port K Pins Alternate Functions

Port Pin	Alternate Function
PK7	ADC15/PCINT23 (ADC Input Channel 15 or Pin Change Interrupt 23)
PK6	ADC14/PCINT22 (ADC Input Channel 14 or Pin Change Interrupt 22)
PK5	ADC13/PCINT21 (ADC Input Channel 13 or Pin Change Interrupt 21)

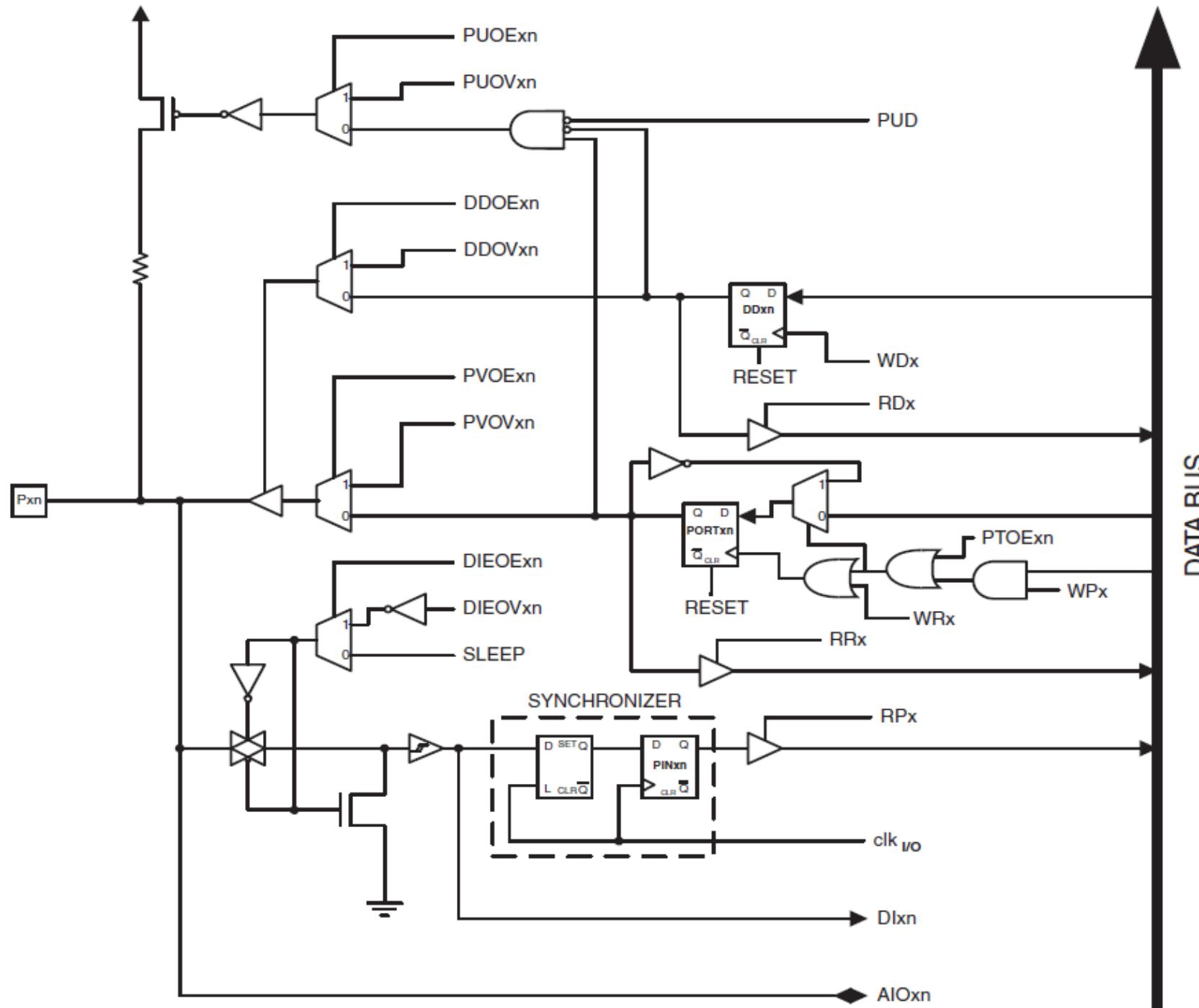
Port Pin	Alternate Function
PK4	ADC12/PCINT20 (ADC Input Channel 12 or Pin Change Interrupt 20)
PK3	ADC11/PCINT19 (ADC Input Channel 11 or Pin Change Interrupt 19)
PK2	ADC10/PCINT18 (ADC Input Channel 10 or Pin Change Interrupt 18)
PK1	ADC9/PCINT17 (ADC Input Channel 9 or Pin Change Interrupt 17)
PK0	ADC8 /PCINT16 (ADC Input Channel 8 or Pin Change Interrupt 16)

13.3.11 Alternate Functions of Port L

The Port L alternate pin configuration is as follows:

Table 13-33. Port L Pins Alternate Functions

Port Pin	Alternate Function
PL7	—
PL6	—
PL5	OC5C (Output Compare and PWM Output C for Timer/Counter5)
PL4	OC5B (Output Compare and PWM Output B for Timer/Counter5)
PL3	OC5A (Output Compare and PWM Output A for Timer/Counter5)
PL2	T5 (Timer/Counter5 Clock Input)
PL1	ICP5 (Timer/Counter5 Input Capture Trigger)
PL0	ICP4 (Timer/Counter4 Input Capture Trigger)



PUOExn: Pxn PULL-UP OVERRIDE ENABLE
PUOVxn: Pxn PULL-UP OVERRIDE VALUE
DDOExn: Pxn DATA DIRECTION OVERRIDE ENABLE
DDOVxn: Pxn DATA DIRECTION OVERRIDE VALUE
PVOExn: Pxn PORT VALUE OVERRIDE ENABLE
PVOVxn: Pxn PORT VALUE OVERRIDE VALUE
DIEOExn: Pxn DIGITAL INPUT-ENABLE OVERRIDE ENABLE
DIEOVxn: Pxn DIGITAL INPUT-ENABLE OVERRIDE VALUE
SLEEP: SLEEP CONTROL
PTOExn: Pxn, PORT TOGGLE OVERRIDE ENABLE

PUD: PULLUP DISABLE
WDx: WRITE DDRx
RDx: READ DDRx
RRx: READ PORTx REGISTER
WRx: WRITE PORTx
RPx: READ PORTx PIN
WPx: WRITE PINx
 $\text{clk}_{\text{I/O}}$: I/O CLOCK
Dlxn: DIGITAL INPUT PIN n ON PORTx
AIOxn: ANALOG INPUT/OUTPUT PIN n ON PORTx